Research Report

Woods
Hole
Oceanographic
Institution

Summer 2000

Irene Zeldenrust 0554944

Contents

Preface

In the spring of the year 2000, it was agreed between John Woodside (Research & Teaching at the Vrije Universiteit – Amsterdam – The Netherlands), Ralph Stephen (Senior Scientist at WHOI (Woodshole Oceanographic Institution) and myself Irene Zeldenrust (undergraduate student at the Vrije Universiteit) that I would do an apprenticeship at WHOI in the summer of 2000. For three months, I studied the data of broadband seismometers located on the oceanbottom near Hawaii. The initial goal was for me to use shear-wave resonance's to better constrain the sediment thickness at the H2O site, because the available seismic reflection data were of poor quality and could not provide enough certainty. The reason for constraining the sediment thickness was that Ralph Stephen and others were submitting a drilling proposal for ODP to drill a re-entry hole to install a borehole observatory. If the sediment thickness is less than 100m, it can be a problem to drill a re-entry hole. Because the H2O data only became available to us in the last month of my stay, I used the first 9 weeks to become familiar with the seismic borehole data and MATLAB by working with the OSN data. This involved trying to distinguish patterns in the OSN data resulting from tilt noise and to filter them out of the signal using coherence between the channels to calculate the transferfunctions. These analyses where done using MATLAB M-files. So my research can be divided in two parts:
1. Studying tilt-noise using data from OSN and H2O sites
2. Studying ocean seismo-acoustic noise resonance's

In the autumn of 2000 I did some more work on the H2O data concerning shear wave resonance's, because I didn't feel I had enough information to write a report. Ralph Stephen and me submitted an abstract for the AGU Fall Meeting on this subject, and we were invited to do a poster presentation. I spent a lot of time on preparing for this meeting and generating figures. I worked by logging onto the WHOI computers using a Secure Shell connection. It took a long time before the connection was established because the computer that I had worked on at WHOI crashed. When a connection was made I soon found out that as a result of everything going back and forward encrypted it is a slow way to work. But it did work and in the end and I was able to analyse another day of H2O data and do some other things I thought would make this research more complete. I learned a lot about working with Unix and MATLAB in addition to gaining experience in working with seismic data from broadband seismometers.

I've tried to write this report in a way that it will also be understandable for geologists with little knowledge of geophysics, since I am myself a mere structural geologist, getting involved in some very interesting, but also very geophysical matters. The subject of my research is far too complicated and broad for me to cover fully in a few months. There are many interesting aspects that I could not study further although I would've liked it. But I hope to present a clear account of the things I did do.

Introduction

Data from seafloor observatories are generally of poorer quality then continental site data because the sea surface is an important and local source of broadband noise. This noise is the result of wind and wave action through direct forcing at long periods and by non-linear coupling to elastic waves at short periods. By analysing the noise and looking for patterns we sometimes can constrain the noise source and subtract it from the spectrum and thereby improve the quality of the data. In another case we can use patterns in the spectrum to derive information on the structure of the ocean bottom.

This report will present an analysis of broadband noise at two locations in the Pacific. The first location is ODP Site 843B, and is situated about 225km south-west of Oahu at a waterdepth of 4407m. Between February and May 1998, the Ocean Seismic Network Pilot Experiment (OSNPE) acquired over 115 days of broadband borehole seismic data at a sampling rate of 20 samples/second. The other location is situated near 28N latitude, 142W longitude (about halfway between Hawaii and California) at a waterdepth of 5000m. In September 1998, a permanent deep ocean scientific research facility – the Hawaii-2 Observatory, or H2O – was installed on a retired AT&T submarine telephone cable recording broadband seafloor seismic data at 160 samples/second.

The analysis has been done using MATLAB, and concentrates on two bands within the seismic noise spectrum.

- The Long-period noise band (0.03Hz – 0.1Hz), in which tilting of the instrument package caused by currents results in high noise levels on the horizontal channels. By determining the coherence between horizontal and vertical channels and calculating transferfunctions the tilt noise can be subtracted and the S/N ratio improved. This analysis is based on the findings of *Wayne C. Crawford and Spahr C. Webb* in their report "Identifying and removing tilt noise from Low frequency (<0.1 Hz) seafloor vertical seismic data".
- The Short-period noise band (1Hz – 70Hz) in which the peaks of S-wave resonance's can be distinguished. These peaks are highest on the horizontal channels and result from resonance's of S-waves of vertical polarisation in the upper sediment layer, exited by the diffuse infrasonic sound field in the water. By identifying these peaks sediment thickness may be constrained. This analysis is based on the findings of *O.A.Godin and D.M.F.Chapman* in their paper "Shear-speed gradients and ocean seismo-acoustic noise resonance's". Apart from looking at only the first five resonance peaks, as in the above-mentioned paper, this analysis also involves peaks at higher frequencies up to 70 Hz.

The conclusions drawn can give us more insight into the possibilities created by the use of broadband seismometers on and under the seafloor, how to expand the global seismic network and thereby deepen our knowledge of the Earth's interior.

General Scientific Background

In past years marine geologists and geophysicists have been trying to expand our seismic network that is now mainly located on land, to the oceanbottom. Because of the gaps (oceans) in the global seismic network [*Wysession*, 1996] we currently have a biased and uncomplete image of the Earth's interior [*Romanowicz*, 1991]. Therefore deep-sea observatories are needed complementary to land stations. **The ocean surface (the waves) is the primary seismic noise source for both terrestrial & ocean floor observatory sites. Being closer to this source, the deep-sea observatories are noisier than good continental sites.**

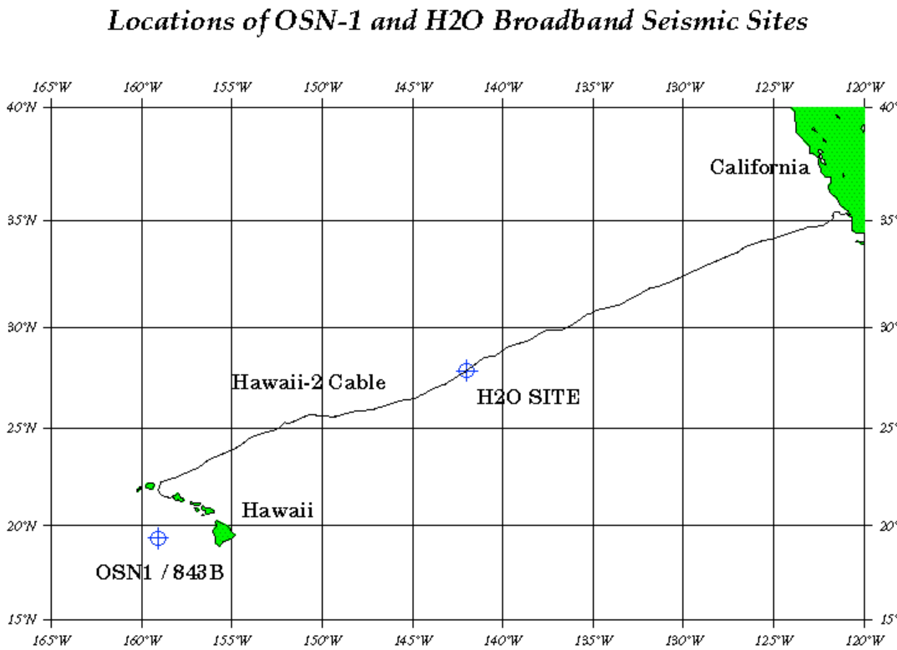


Fig.A1: The Ocean Seismic Network Site (OSN-1) is 225km southwest of Oahu in 4407m water depth. The Hawaii-2 Observatory (H2O) is half-way between Hawaii and California on the retired Hawaii-2 telecommunications cable and is in 4970m water depth.

✓ The OSN Site

The OSN Pilot Experiment was carried out in ODP Hole 843B which was drilled on ODP Leg 136 in February-March, 1991. The primary goal was to learn how to make high quality broadband (0.003-5Hz) seismic measurements on the sea floor in preparation for extending the Global Seismic Network to the ocean basins. The results from the three oceanic sensors can be compared with each other and with other seismometer systems on or near the Hawaiian Islands, for a range of ambient noise conditions and earthquake sources.

The geological setting

Between February and May 1998, the Ocean Seismic Network Pilot Experiment (OSNPE) acquired over 115 days of broadband borehole seismic data at ODP Site 843B in 4407m water depth off Oahu. The borehole seismometer was clamped in casing within the upper 6m of igneous basement beneath 242m of sediment.

The research facility

On the cruise three broadband seismic instruments were deployed: a borehole seismometer, a sensor buried surficially in the sediments, and a sensor resting on the seafloor. The borehole seismometer was a Teledyne KS54000 similar to the sensors used in the global IRIS/IDA and GSN networks. It was placed in the borehole using the MPL/JOI Wireline Re-entry System. The seafloor and shallow buried sensors were Guralp CMG-3T three component seismometers. In addition three conventional Ocean Bottom Seismometers with 1Hz geophone sensor, differential pressure gauges, a conventional hydrophone and a current meter were deployed. All three of the broadband instruments were designed to record continuously and autonomously on the seafloor for at least three months [*Stephen et al.,*1998].

The data

In addition to over fifty earthquake events that were observed, ranging from a 4.5Mb event at 44° epicentral distance to the 7.9Mw Balleny Islands earthquake at 91 epicentral distance, this data set provides an opportunity to study the time dependence of ambient noise in a borehole in the deep sea over a four-month duration. The ambient noise behaviour falls into four distinct frequency bands (described later on in this section). In the infra-gravity band,
1-10mHz, where ambient noise levels are greater than levels on the co-located buried sensor, there is a strong tidal frequency dependence indicating that water flow in the hole, past the instrument, may be exciting installation noise. The largest amplitude changes occur in the noise notch, 10-50mHz, where vertical component noise varies from the quietest levels observed worldwide -185dB to levels above 120dB after the Balleny Islands earthquake. The microseism band, 50-300mHz, is characterised by three peaks. Levels of the single frequency microseism peak at 60mHz can increase 60dB after a large earthquake. The levels of the two double frequency micro-seism peaks, one each from distant and local sources, is much less variable (less than 20dB) and is related to sea state. The short period band (or HOLU spectrum), 300mHz to 7.5Hz, consists of a set of peaks that correspond to shear modes in the seafloor which are excited by local sea state. Above 5Hz there is a weak tidal dependent effect, primarily on the vertical component, which could be related to bottom currents washing against the re-entry cone.

✓ The H2O Site

In September 1998, a permanent deep ocean scientific research facility - the Hawaii-2 Observatory, or H2O - was installed on a retired AT&T submarine telephone cable that runs between Oahu, Hawaii and the California coast. The facility consists of a sea floor junction box and scientific sensors located in 5000m of water near 28N latitude, 142W longitude, or about halfway between Hawaii and California. The H2O site is located at a point on Earth's surface where there is no land for about 2000 km in any direction. For this reason, it is a high priority site for the Ocean Seismic Network (OSN) component of the Global Seismic Network (GSN), and serves as the first operational OSN station. While the present H2O seismometer utilizes a buried broadband sensor, the Ocean Drilling Program (ODP) is scheduled to drill a re-entry borehole at the H2O site at the end of 2001for subsequent installation of a down hole seismometer.

The geological setting

The lithosphere west of 140°W in this area was formed between the Pacific and Farallon plates under normal spreading conditions, but at a fast half rate of 7 cm/y [2,3]. The crustal age based on magnetic lineations is about 45 Ma (iso-chron 20) or mid-Eocene. The regional physiography is one of abyssal hills with a nominal but variable 50-100 m cover of terrigeneous clay sediment. The local relief around the H2O junction box is quite subdued; a Jason survey for 1 km around that point reveals no rock outcrops and very gentle relief of a few tens of meters on a smoothly sedimented bottom.

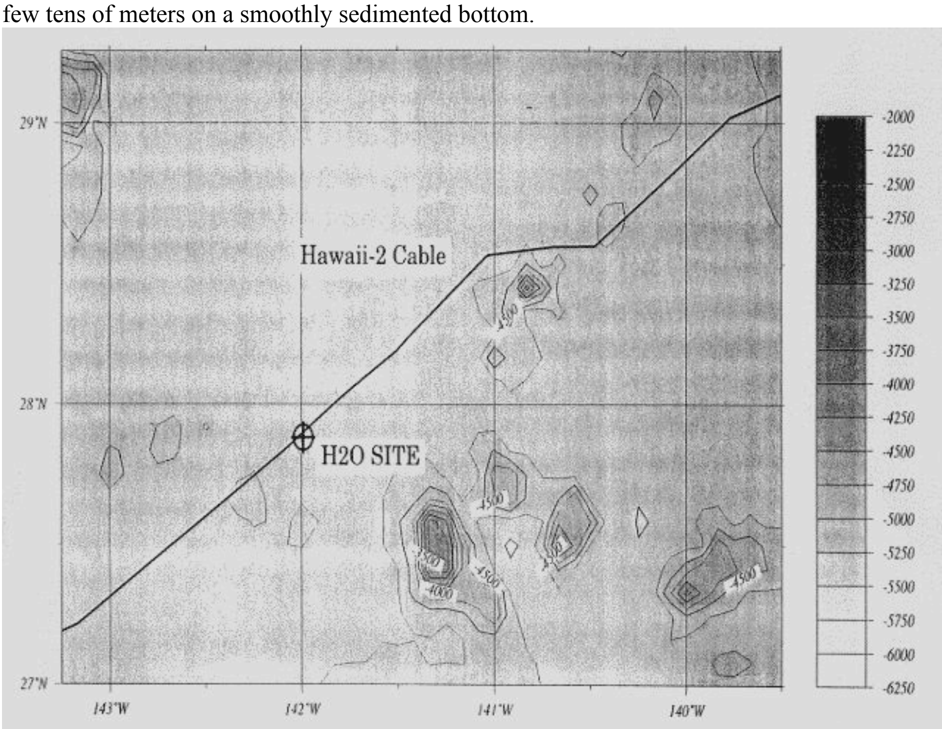


Fig.A2: Bathymetry along Hawaii-2 cable.

It is known from earlier drilling (DSDP, Legs 5 and 18) that there are chert layers in this part of the Pacific. On the 3.5KHz records there is a clean single pulse followed 10 msec later by a diffuse event. The clean event is interpreted as the sea floor and the diffuse event is the chert layer. Ten mseconds of TWTT corresponds to about 8m thickness of soft sediments. Nothing coherent is seen with 3.5KHz below the chert layer. This was also the experience in the 1960 surveys where “acoustic basement” was chert. There is a continuous mid-sediment reflector at about 0.3 secs or about 25m depth, which does not correspond to the chert layer identified on the 3.5 records. If the diffraction events at about 0.6 secs in the SCS data are interpreted as occurring at the sediment-basement boundary we get a very uniform sediment thickness of about 50m. This may get as thick as 75m in some areas but in no area 100m of sediment was identified. The following figures show unmigrated and migrated seismic lines in the vicinity of the H2O Observatory.

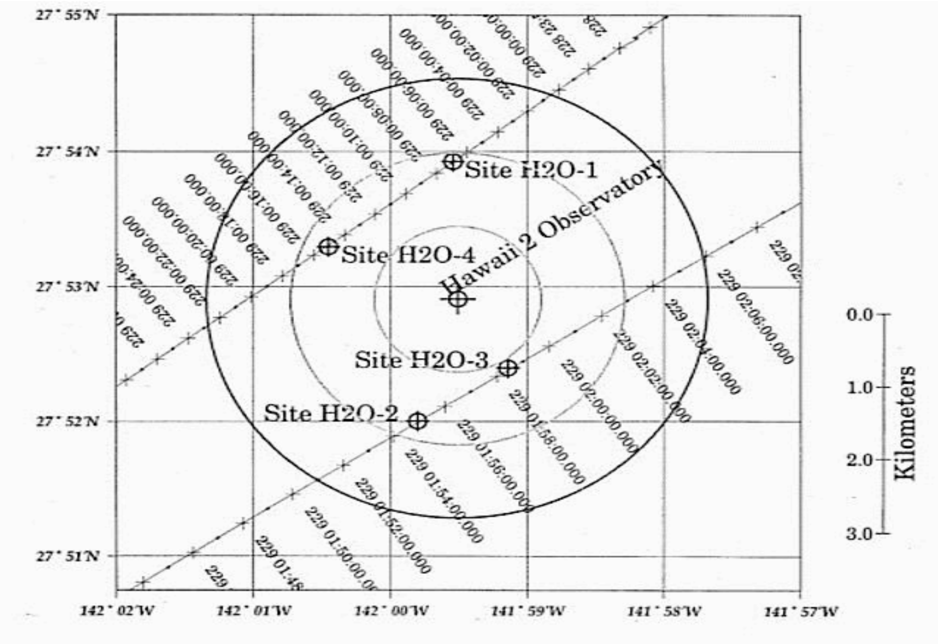


Fig.A3: Survey lines in the vicinity of the H2O Observatory, annotated in Julian day and time.

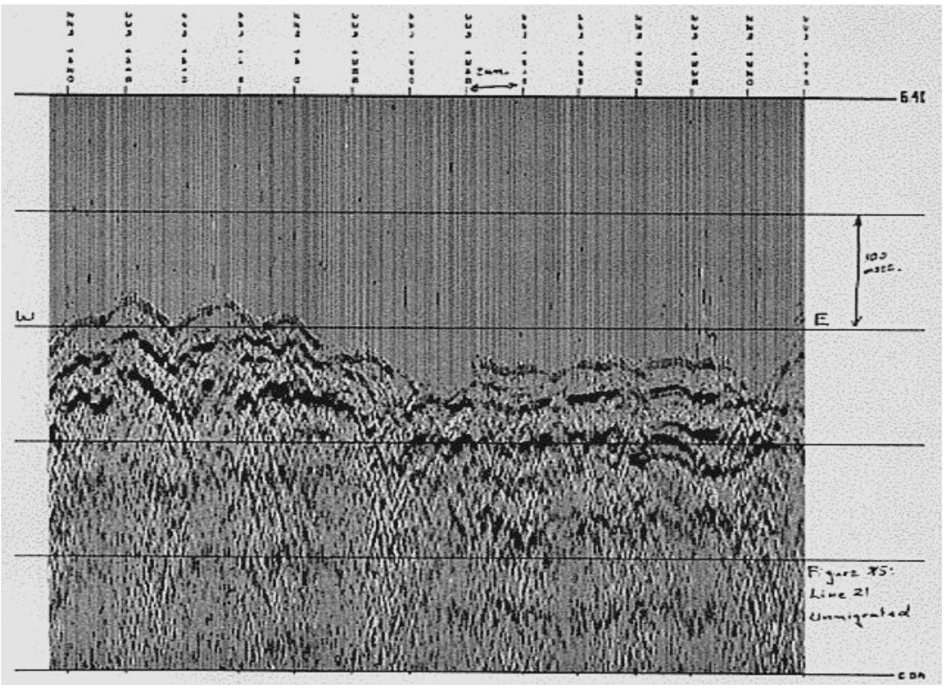


Fig.A4: Seismic line 21 North of the H2O site. Unmigrated.

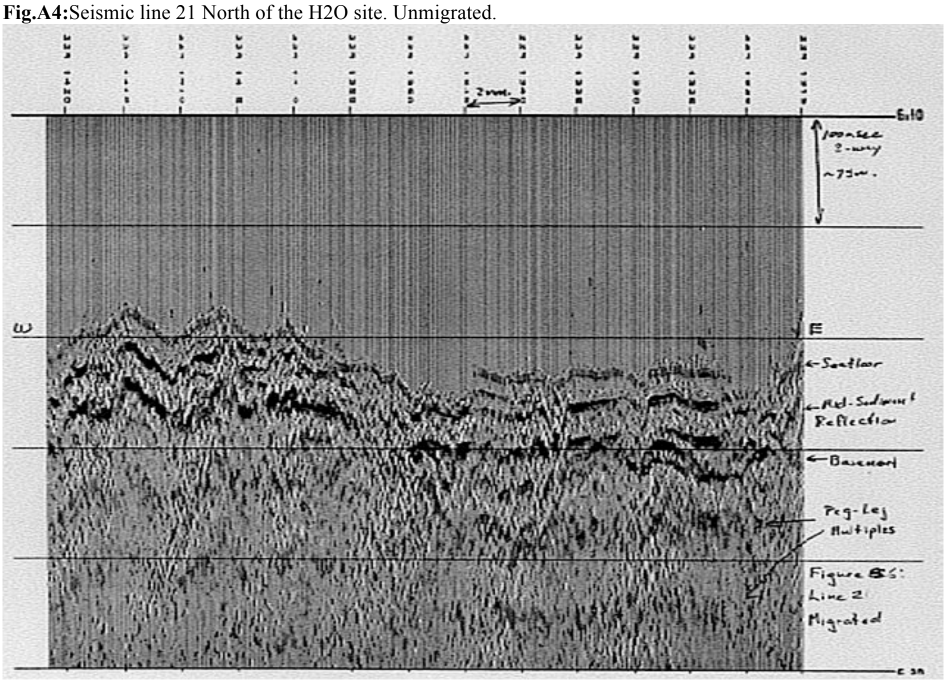


Fig.A5: Seismic line 21 North of H2O site. Migrated.

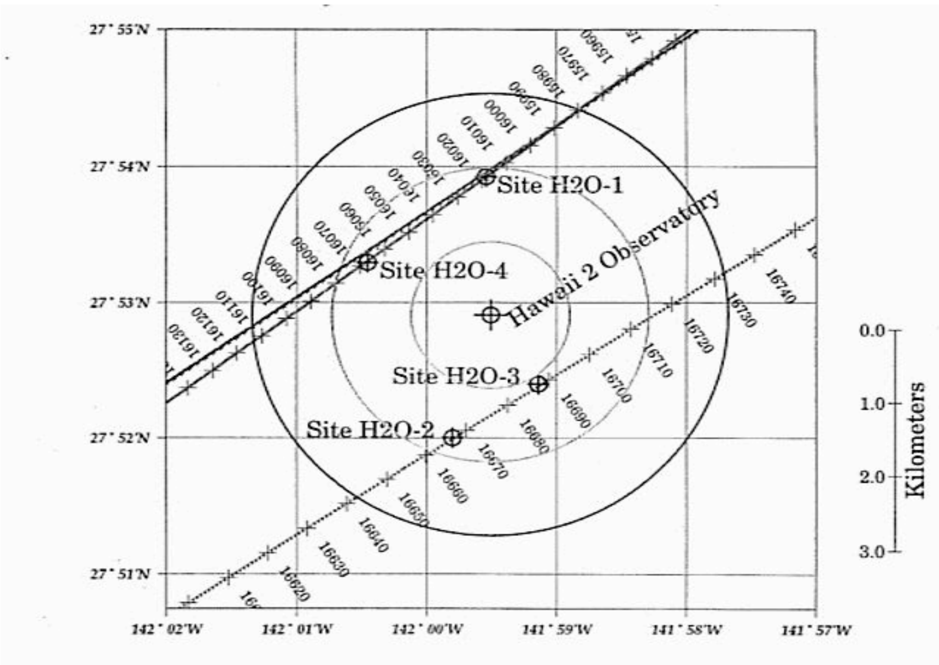


Fig.A6: Survey lines in the vicinity of the H2O Observatory, annotated in SCS shot numbers.

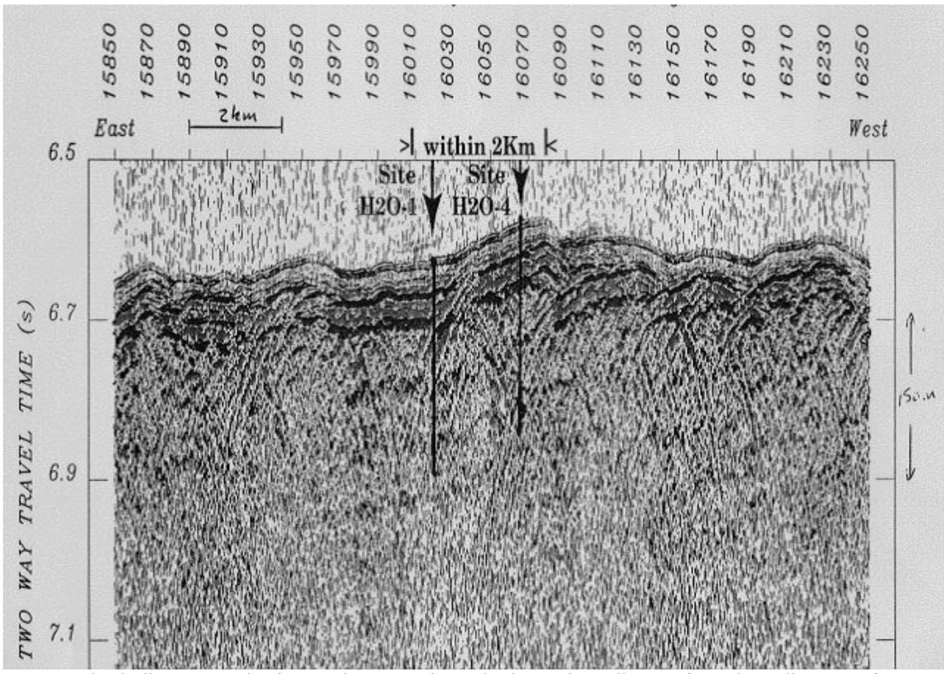


Fig.A7: Seismic line 14 North of H2O site. Not migrated. Shows the sediments from the sediment surface to a depth of 500m.

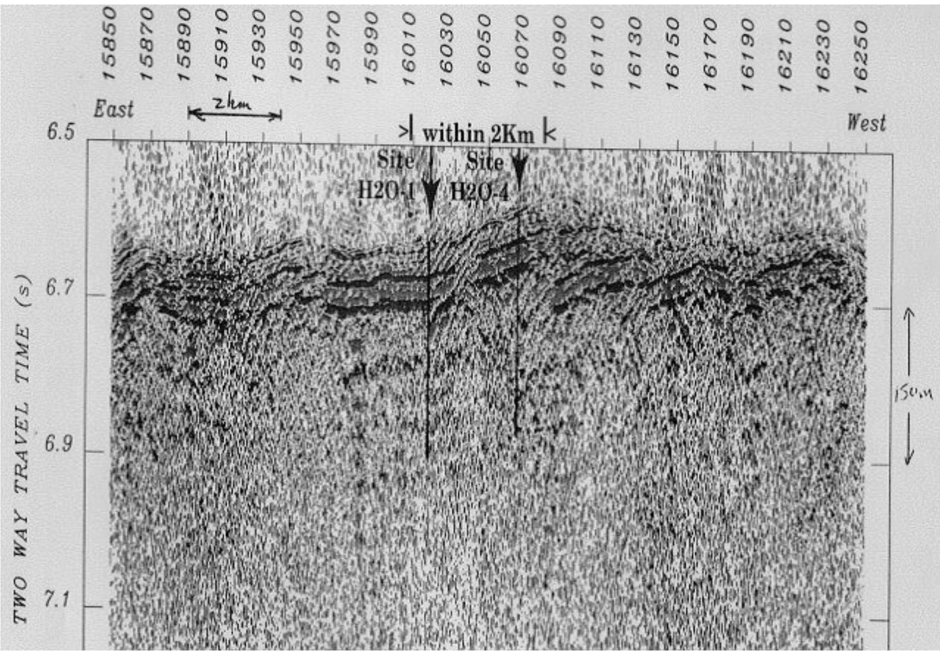


Fig.A8: Seismic line 14 North of H2O site. Migrated. Shows the sediments from the sediment surface to a depth of 500m.

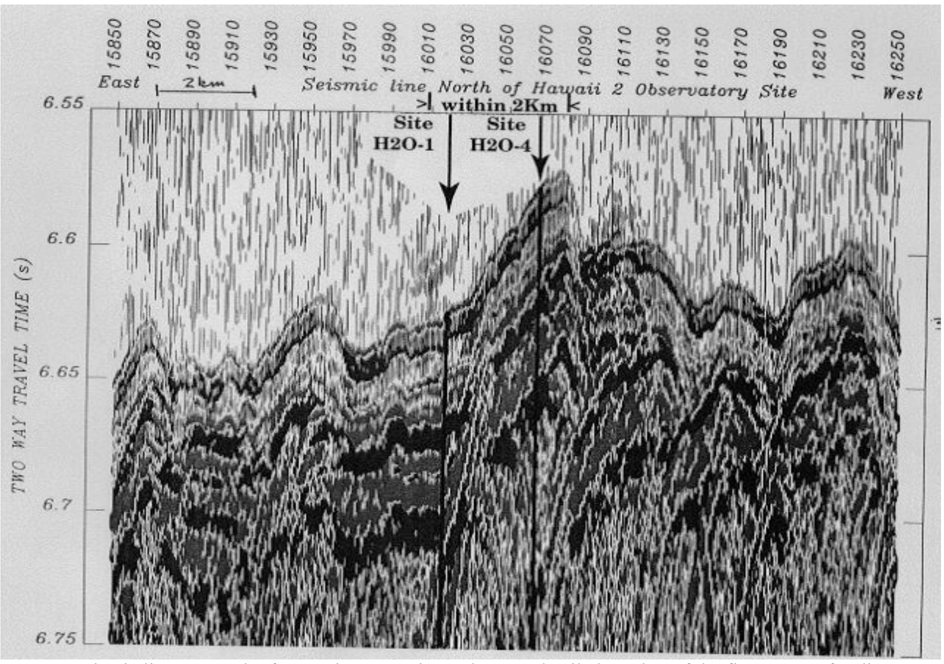


Fig.A9: Seismic line 14 North of H2O site. Not migrated. More detailed version of the first 110m of sediment

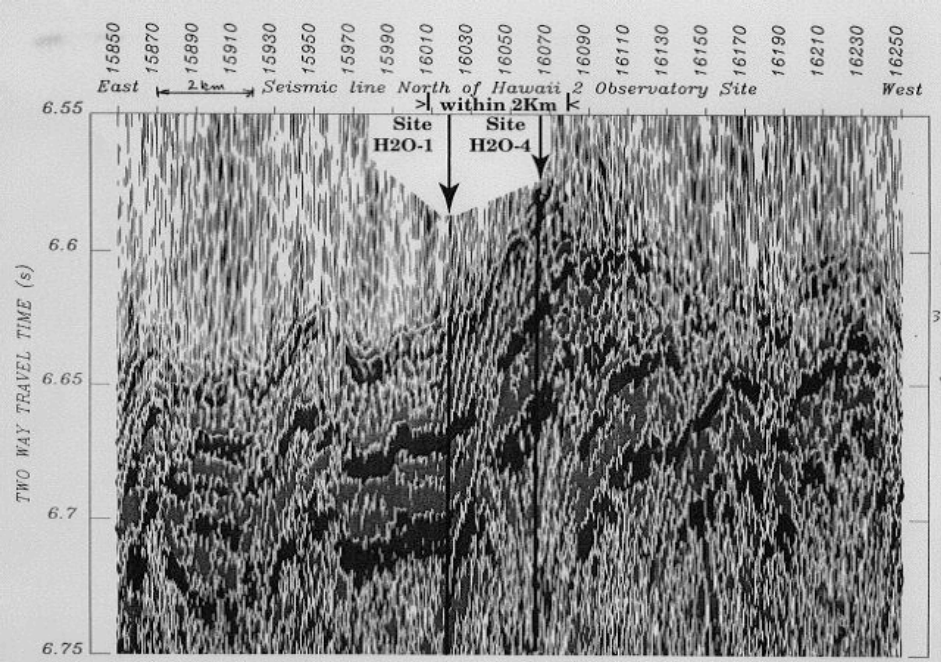


Fig.A10: Seismic line 14 North of H2O site. Migrated. More detailed version of the first 110m of sediment

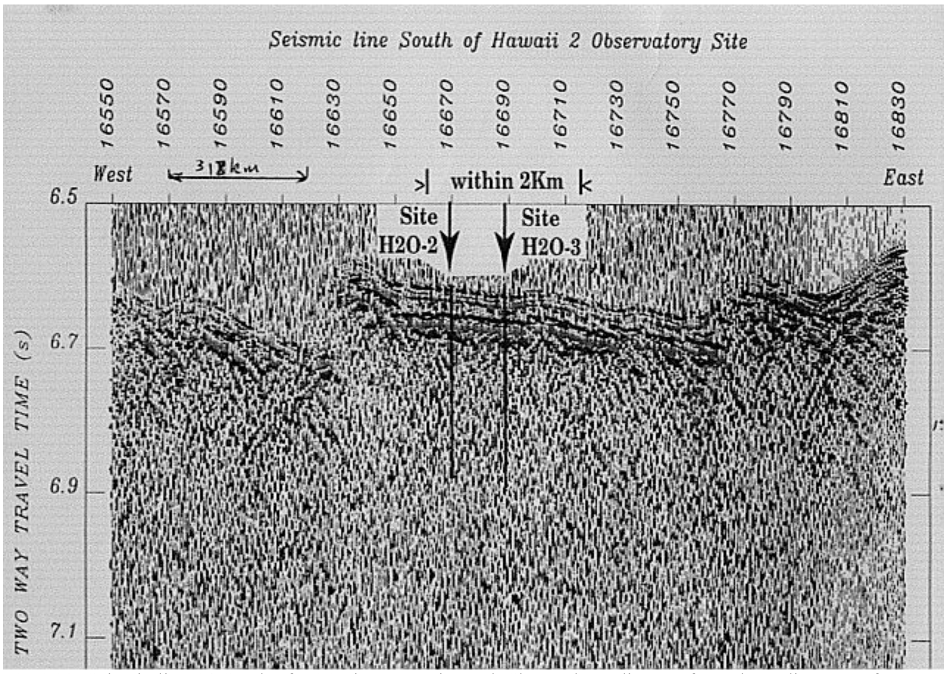


Fig.A11: Seismic line 14 South of H2O site. Not migrated. Shows the sediments from the sediment surface to a depth of 500m.

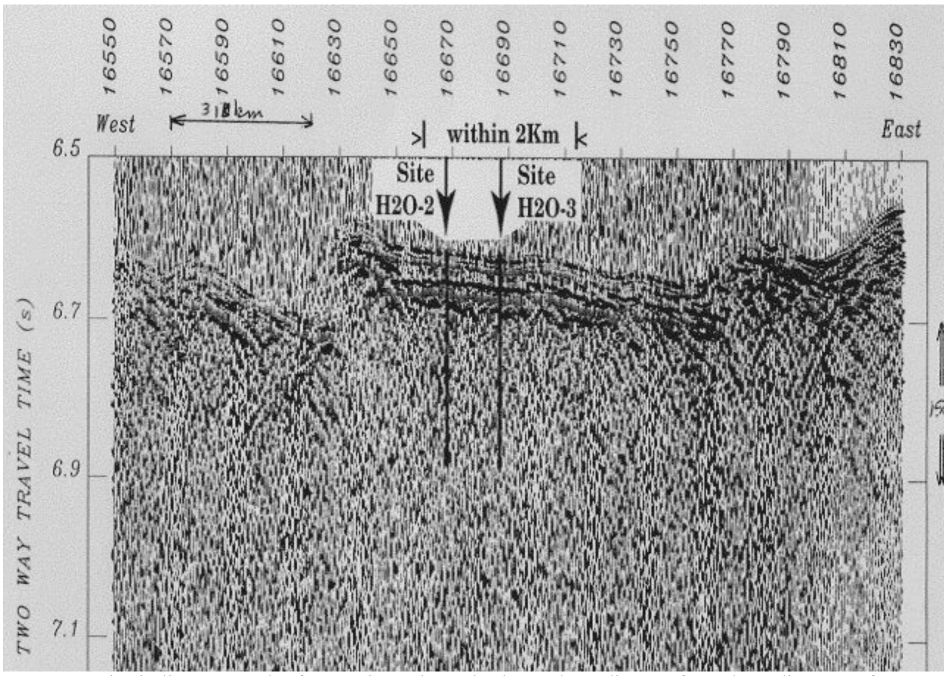


Fig.A12: Seismic line 14 South of H2O site. Migrated. Shows the sediments from the sediment surface to a depth of 500m.

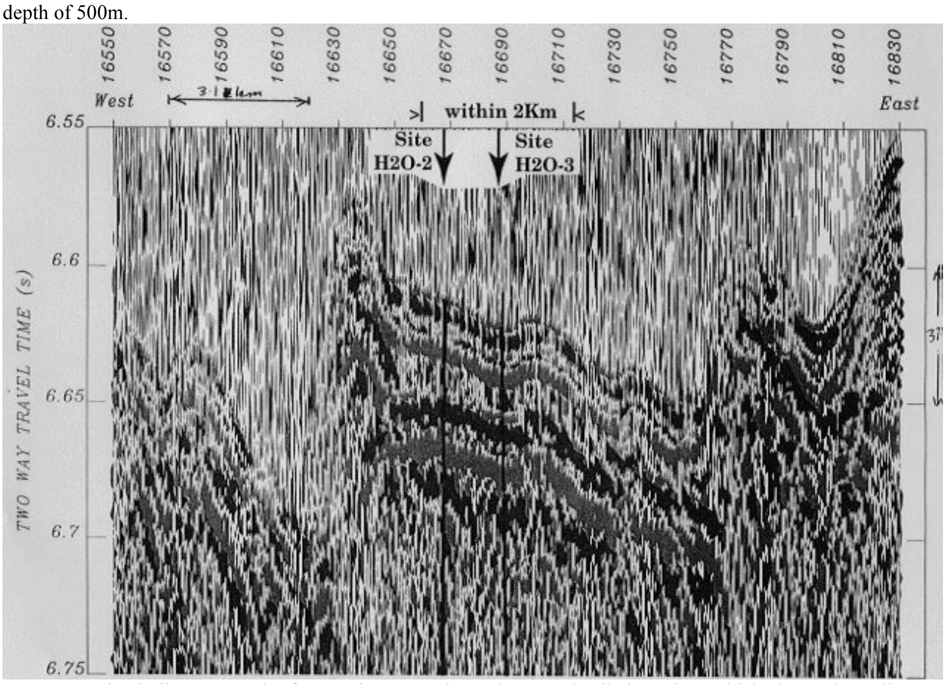


Fig.A13: Seismic line 14 South of H2O site. Not migrated. More detailed version, which shows the sediments from the sediment surface to a depth of 110m.

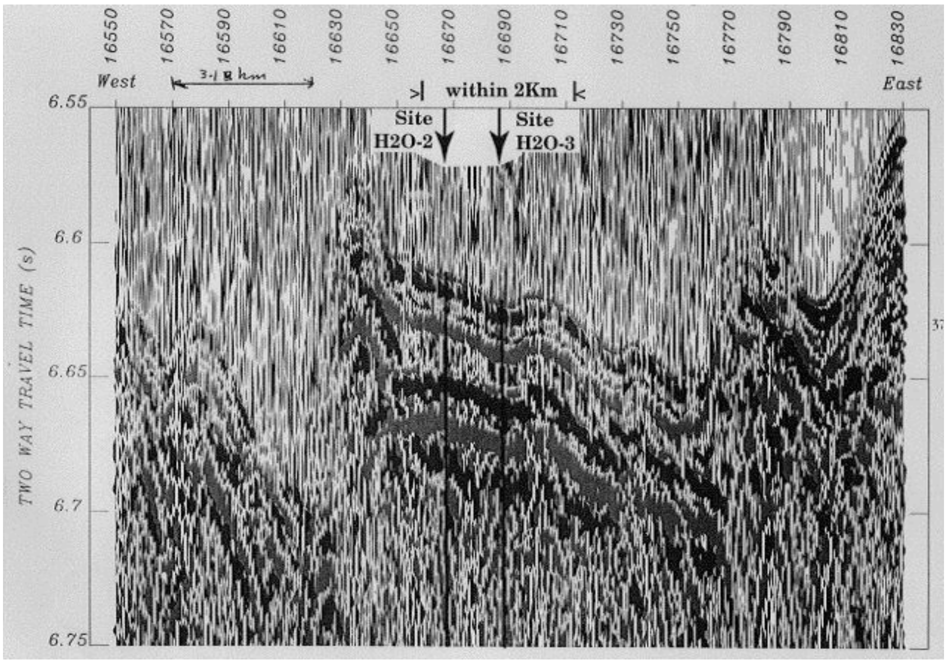


Fig.A14: Seismic line 14 South of H2O site. Migrated. More detailed version, which shows the sediments from the sediment surface to a depth of 110m.

Because of the proposal to drill one or more holes at the H2O site, there needs to be a discussion with ODP drilling engineers about their thoughts on drilling a re-entry hole in 50-75m of sediment [from the *ODP-Drilling Proposal*, 1997].

The research facility

The junction box derives 400 watts of power from the cable to power both itself and user scientific instruments, and provides two-way communication through 8 digital ports with wet-mateable connectors. Instruments may be connected to the junction box using a remotely operated vehicle (ROV). Initial instrumentation at the H2O site includes a broadband three-component seismometer (Guralp CMG-3), orthogonal 4.5 Hz geophones, a broadband hydrophone, and a pressure sensor. The seismic sensors are buried in a caisson just below the ocean floor, and data are sampled in high and low gain channels at 160 samples per second. [*Duennebier et al.,* 2000] H2O is also the first seafloor station in the Global Seismographic Network. The H2O system is connected to the Internet via the cable terminus on Oahu and the University of Hawaii. Data from H2O are publicly available through the IRIS Data Management Centre. This site provides marine scientists with a new opportunity to deploy and operate remote instrumentation in the middle of the ocean.

The data

The H2O Observatory has been acquiring real time data from the ocean floor since October 2000. Since this date it has recorded most earthquakes larger than magnitude 6 world-wide. Data are routinely displayed in two frequency bands, 0.02-0.09 Hz and 5 Hz high pass, avoiding the strong microseism peak. At frequencies above 5 Hz, shipping, whales, and earthquake T-phases are common signals, while teleseismic earthquakes are visible mainly on the low frequency records. [*Duennebier et al.,* 2000]

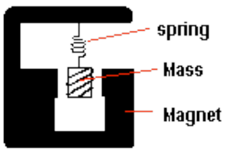
✓ The Signal

A signal consists of:
- Different sine and cosine waves, thus it has a certain frequency content (many high frequencies and many low frequencies).
- Events, all signals that result from big earth movements (explosions, earthquakes). The amplitude depends on the distance.
- Noise, all signals that do not result from the above mentioned like waves, currents, humans, and ships. This noise can be removed if the wavelength is found and the amplitude.

✓ The Seismometer

What is an Ocean Bottom Seismometer?

Seismometers acquire sound information through the earth. Acoustic waves (sound) travel in the form of very small motions which advance through the earth with time. The seismometer



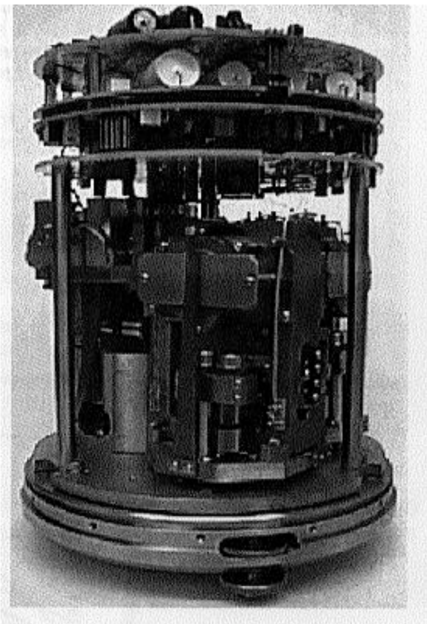
(or geophone) is a detector that is placed in direct contact with the earth to convert very small motions of the earth into electrical signals, which are recorded digitally. Each seismometer consists of a mass, which contains an electrical coil that is suspended by a spring between the poles of a magnet. When the earth moves, the magnet and spring support move with the earth. The mass tends to remain stationary, so its motion will lag behind that of the magnet. This relative motion produces a voltage that is proportional to the velocity of this motion. Motions of the earth, combined with precise timing and location information for the sound source and the receiver (the seismometer), can provide details of the velocity and the geometry of the earth structure. Since the speed of sound is related to the density of the medium, the earth transmits the acoustic wave faster than seawater and deeper layers in the earth transmit the waves faster than shallow ones.

The instrument measures a range of frequencies in three orthogonal directions (Z, H1, H2). It measures these frequencies using electrical coils that are suspended and work on the principle of inertia. H1 and H2 respond to horizontal ground movements like
- *Love waves*. They travel along a surface when there is an increase in S-wave velocity with depth or, if layers are present, the S-wave velocity of the overlying layer is less than below. Ground particles move in a direction normal to the propagation path and in a plane parallel to the surface.
- *Body waves*. They can travel deep inside the Earth. Energy transfer takes place in compressional (P) and shear (S) waves. P-waves are characterised by particle motion in the medium parallel to the direction of propagation. Water-borne P-waves of short (<1s) period are known as T-phases. The passage of seismic energy as S-waves involves particle motion normal to the direction of travel. Shear speed can be polarised in any direction. Since fluids cannot maintain shear stress, S-waves are not observed in water. All body waves are non-dispersive; that is, their speed is independent of frequency.
- *Rayleigh waves*. They are exited when a medium had a free surface. They are characterised by ground motions following retrograde ellipses in a vertical plane containing the direction of propagation.

Z responds to vertical ground movements like
- *Surface waves*. They are confined to the outer parts of the solid earth
- Vertically travelling *body waves*.

Every signal usually has components in all three directions.

At both the H2O and OSN site, the buried broadband seismometer is a *Guralp CMG-3(T)*. Model 3T components are optimised for low noise at long period, and the flat velocity response is 0.01 to 50Hz. The CMG-3T is used in the USGS National Seismograph Network. Model 3ESP has flat velocity response from 0.03 to 50Hz and is used in portable and fixed applications [http://www.guralp.demon.co.uk/shrtform/catalogue1.htm].



✓ The seismic spectrum

The intensity of "noise" at different frequencies in the recordings of an OBS can be shown in a record of the noise spectrum (in the frequency domain) while the intensity of noise through time can be shown in a time record of the acceleration (in the time domain). Figure A15 shows spectra for land, island and seafloor stations. A typical noise spectrum consists of a **long-period** (< 0.1 Hz) and a **short-period** (> 0.1 Hz) band, separated by a **microseism peak** (around 0.2 Hz).

Long-period noise levels are controlled by ocean currents and by deformation of the ocean bottom under infragravity waves. Long-period waves are produced by large ocean storms and travel thousands of kilometres [e.g.,*Snodgrass et al.*,1966]. There have been few long-period measurements of seafloor noise using seismometers, but many long-period pressure measurements.

The short-period noise levels are dominated by energy coupled from ocean waves into acoustic waves and then into elastic waves below the seabed causing ground motions called microseisms. They are the largest noise source for both continental and oceanic stations, but the microseismic peak is more energetic and broader at seafloor stations. The short-period ocean wave-field is primarily determined by wind local to the area.

Island stations have a quieter noise spectrum than adjacent seafloor stations near 1 Hz, but are still noisy in the microseism band [*Zhang and Langston*, 1995; Li et al., 1994; *Peterson,* 1993; *Hedlin and Orcutt*,1989] compared with most continental sites. At some locations high frequency noise at seafloor sites could be comparable to noise levels at island sites [*Hedlin and Orcutt*, 1989], but such quiet conditions are rare in the Pacific.

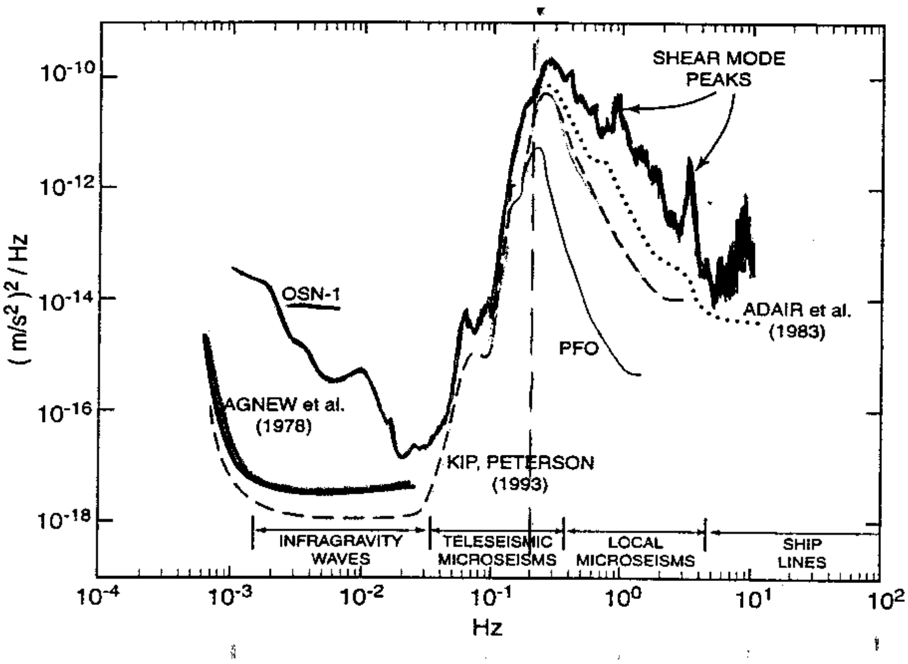


Fig. A15: Vertical acceleration spectra from the seafloor south of Hawaii (OSN-1) [*Webb et al*., 1994]; from Kipapa, Hawaii [from *Peterson*, 1993]; and from a quiet site in California (PFO), with the long-period spectrum from the same site [*Agnew and Berger*, 1978]. The microseism peak at 0.2 Hz divides the spectra into long- and short-period bands.

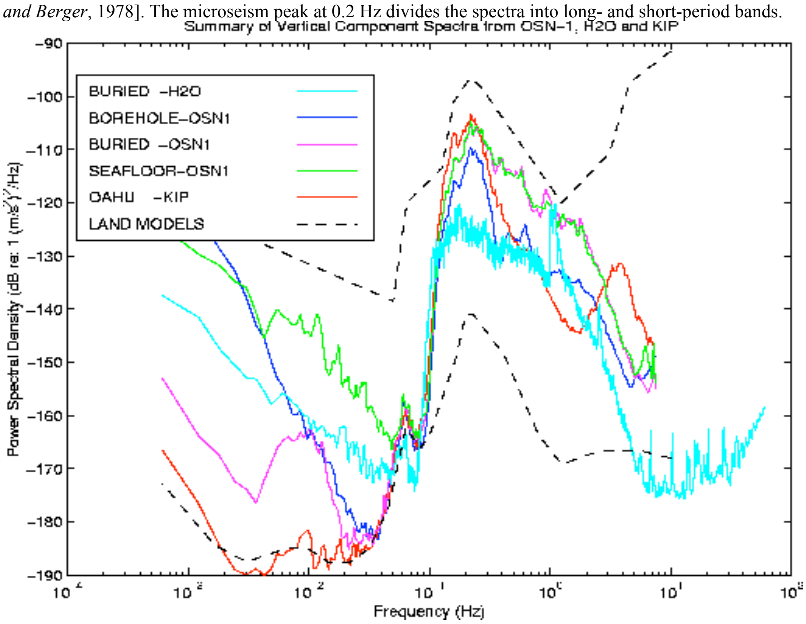


Fig.A16: Vertical component spectra from the seafloor, buried and borehole installations at OSN-1 are compared with the spectra from the buried installation at H2O and the KIP GSN station on Oahu. H2O has extremely low noise levels above 5Hz and near the micro-seism peak from 0.1-0.3Hz. H2O has high noise levels below 50mHz.

Otherwise H2O levels are comparable to OSN borehole and KIP levels. The sediment resonances at H2O near 1 and 3 Hz are very prominent.

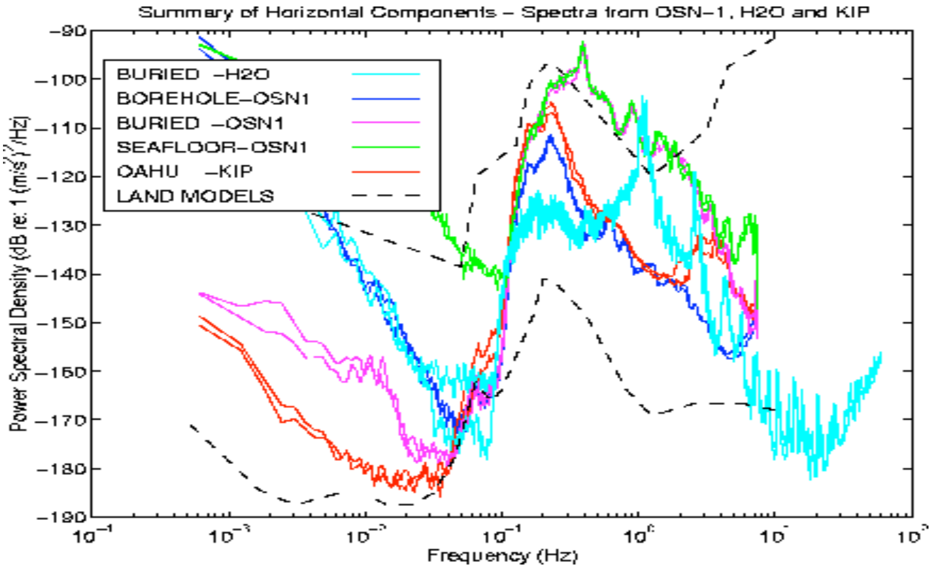


Fig.A17: Horizontal component spectra from the seafloor, buried and borehole installations at OSN-1 are compared with the spectra from the buried installation at H2O and the KIP GSN station on Oahu. The sediment resonance peaks in the band 0.3-8Hz are up to 30dB louder than background and far exceed the microseism peak at 0.1-0.3Hz. That the resonant peaks are considerably higher for horizontal components than for the vertical component is consistent with the notion that these are related to vertically propagating shear wave multiples.

Moving from left to right in a typical noise spectrum, the noise levels in different frequency bands are now discussed:

✔ **Infragravity waves** (<0.03Hz)

Elevated noise levels at long-period are due to the deformation of the seafloor under low-frequency, freely propagating ocean waves generated by nonlineair processes from the wind, waves and swell [*Webb et al*., 1991]. A 100s-period infragravity wave will have a wavelength of 1 km in 10 m of water, increasing to 4.5 km in 200m depth and to 20 km in deep water. These infragravity waves control oceanic *vertical seismic noise levels* at periods > 30s and can raise noise levels by as much as 40 dB. But noise levels are still low enough to provide good SNR for Rayleigh waves to periods as long as 200s. Amplitudes of these waves in deep water are small, less than 1 cm in wave height in the band from 0.002 to 0.03 Hz.

The effect of tilting on the *horizontal noise levels* due to the deformation signal is small because infragravity wave wavelengths are large. Horizontal displacements have therefore not been observed at deep seafloor sites because of measurement difficulties; the flow noise is usually orders of magnitude larger. But when sensors are installed below the seafloor to remove flow noise, infragravity waves will control long-period seismic noise levels for *both horizontal and vertical components*.

The infragravity wave signal remains significant to depths of many kilometres below the seabed depending on frequency, and elastic parameters as a function of depth (Figure A18).

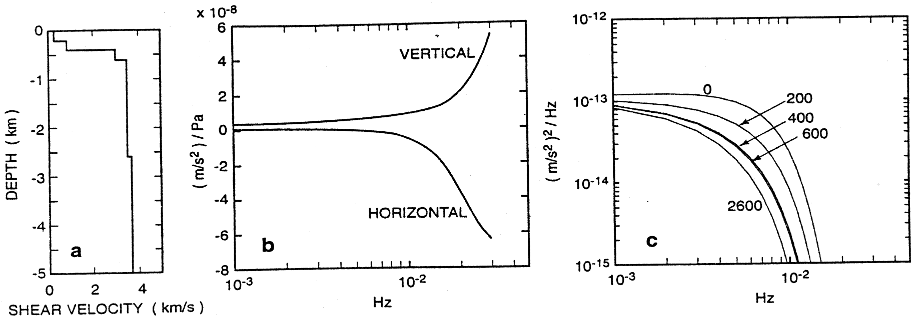


Fig. A18: a) shear velocity in a model of sedimented ocean crust. b) transfer functions between seafloor pressure and seafloor acceleration due to infragravity waves. c) predicted acceleration spectrum due to infragravity waves at five depths in a model consisting of 400m of sediment over basalt.

The extend to which ocean bottom is deformed by infragravity waves depends on the shear strength of the oceanic crust. The shorter-period waves deform only the shallow parts of the crust, while the deformation under longer (lower frequency) waves reaches through the entire crust. In the Pacific, the energy of infragravity waves is high and varies little with time.

Measurement of the transfer function between pressure and the displacement under loading of the ocean waves can be used to determine the elastic parameters below a site and to probe the Earth.

Origin and climatology of infragravity waves

There are at least *two mechanisms* that drive infragravity waves:

- **At frequencies < 0.002 Hz**, the pressure spectrum varies inversely with the square of frequency [*Filoux,* 1980, 1983; *Chave et al*., 1992] but apart from this, it is remarkably constant in amplitude with no apparent seasonal cycle. The waves in this band may be driven directly by atmospheric pressure fluctuations.
- **At frequencies > 0.002 Hz**, infragravity wave amplitudes in the N.Atlantic can be directly connected to the short-period ocean wave energy around the Atlantic basin [*Webb et al*., 1991]. Infragravity wave energies varied by more than 25 dB during an 80-day interval. It is this relationship between short- and long-period waves that makes it possible to predict infragravity wave energies, and therefore long-period seismic noise, at any site within the oceans.

Because most long-period energy is trapped within 1 km of the coastline (called "surf beat"), long-period seismic measurements on the seabed near the coast can be 30 dB noisier than deep seafloor observations.

In deep water, infragravity waves are only slightly attenuated [*Lighthill*, 1979], so that waves travel across the ocean basins essentially unaltered, following approximately great circular paths.

North Atlantic sites have been found to be more variable and between 10 and 30 dB quieter than sites in the Pacific [*Webb et al*., 1991; *Babcock et al*., 1994]. The convoluted shape of the North Atlantic basin shields the central basin from infragravity waves from the Arctic and Antarctic regions, resulting in much greater variability and lower longwave energy than in the Pacific [*Webb et al*., 1991]. A wider average shelf may also lead to larger attenuation and a less energetic infragravity wave climate in the North Atlantic. Storms in the NH in winter and in the SH in the summer, as well as typhoons in both seasons, act to maintain an energetic infragravity wave climate throughout the Pacific. Therefore the Pacific generally has higher noise levels than the Indian Ocean or the North Atlantic.

(An interesting thought: the ice-covered seas that surround Antartica in winter could provide good long-period sites at the seafloor if a method to deploy instruments beneath the ice can be established.)

✓ The "noise notch" (0.03 – 0.1 Hz)

This band is important to marine seismology because it provides a **low-noise window** for the detection of Rayleigh waves and long-period body waves.

The *single-frequency microseism peak near 0.07 Hz* (T is approx. 14 s) is a permanent feature of spectra from either the seafloor or from land, although the peak can be partly obscured in Pacific seafloor measurements, by the side of the *main (double frequency) microseism peak at 0.2 Hz*. The origin of the single-frequency peak is loosely associated with direct transfer of ocean wave energy into elastic waves trough non-linear coupling of waves and bathymetry [*Hasselman,* 1963], universality of this peak suggests global sources. It propagates primarily as fundamental mode Rayleigh waves [*Lacoss et al,*.1969].

Below the single-frequency peak the *vertical noise level* appears to be determined by instrumental noise, except for a few sites on the deep seafloor where currents strongly affect the noise levels [*Webb,* 1988]. *Horizontal noise levels* are invariably controlled by currents. Apart from the above, there is little understanding of either temporal or geographical variability of vertical component noise levels in the noise notch at present.

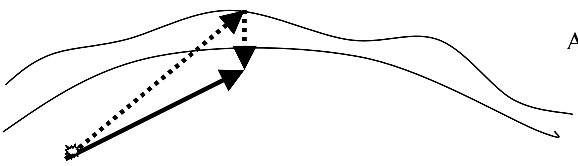
Reverberation problem for long-period body waves

Observations are complicated by water column reverberations hidden by a multipole low-pass digital filter. This filter must be applied to clearly see long-period body wave arrivals below the energetic microseisms and it limits the rise time of arrivals to many seconds.

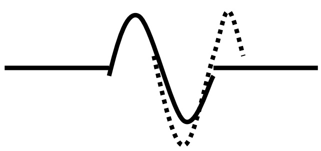
Water column reverberations interfere with the shapes of arrivals. The apparent arrival times shift with the water depth, as the phase of the interference depends on [*Blackman et al*.,1995]:

- filter bandwidth
- water depth

In long-period, filtered records the reverberations merge with the original wave and appear as a single arrival.



And:



Detection limits for long-period body waves

Low noise in the "noise notch" (0.03 – 0.1 Hz) makes it possible to detect long-period (10 – 50s) body waves from moderate size earthquakes ($M_w \geq 5.5$) at teleseismic ranges (> 30°). The best S/N-ratio for body waves occurs at frequencies near 0.06 Hz (17s period), just below the single frequency microseism peak.

Detection is usually done by DGP, 1-Hz geophones or broadband sensors. 1-Hz geophones are sufficiently quiet to allow detection in the 10-20s band, but are noisy at long T because of rapidly rising electronic noise levels. Broadband instruments have the best S/N-ratio but haven't been used much so far.

Horizontal component noise levels at 0.01 Hz can *exceed vertical noise* levels by 30 – 60 dB for instruments on the seafloor because the instruments rock slightly under ocean currents (tilt noise). Despite these high noise levels, there is usually a band between 10-s and 20-s period of sufficiently low noise level to record teleseismic SH arrivals from large earthquakes during quiet intervals.

Ocean floor currents and long-period noise

Horizontal component noise at instruments deployed on the seafloor appears to be controlled primarily by ocean floor currents. **Long-period horizontal noise levels were *30 – 40 dB higher than vertical noise levels* (Figure A19) and appeared to be controlled by currents (and phase of the tide) at a Pacific site [*Sutton et al*., 1965].**

Ocean currents perturb sensors by pushing directly on the sensor package and by deforming the seafloor under the package, both of which cause tilting of the sensor package [*Webb,* 1988]. At a quiet continental site a seismometer may see than 20 dB below the infragravity wave signal vertical accelerations smaller than 10^{-10} m/s^2 in a 0.01 Hz band near 100s period. A fluctuating tilt of 10^{-11} rad will produce a comparable signal on the horizontal components as the acceleration of gravity is rotated into the horizontal components. Forces associated with even quite weak currents are sufficient to cause the tilts necessary to raise long-period horizontal noise levels for any seismometer on the seafloor [*Duennebier and Sutton,* 1995; *Webb*, 1988].

The current noise on the *vertical component* is usually more at frequencies near 0.01 Hz [*Crawford et al*., 1991]. Low noise levels on the vertical component can be achieved only if the vertical component is closely aligned with the true vertical; otherwise tilt noise is rotated into the vertical component, with the noise proportional to the sine of deviation from the true vertical. Tilt noise is also coupled into the vertical if the rotation of the sensor Package lifts the seismometer Vertically, but this term is likely to be important only at higher frequencies, since the acceleration noise from this source is proportional to the second time derivative of the tilt signal, while the misalignment signal is directly proportional to tilt.

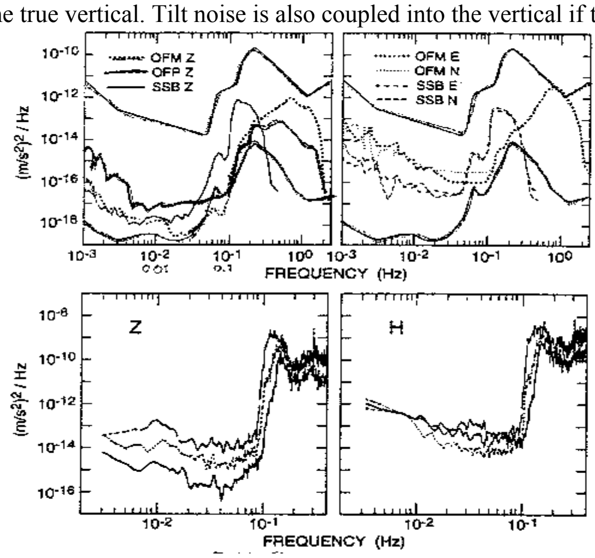


Fig. A19: Long-period spectra (top) from a borehole sensor (OFP) and a seafloor sensor (OFM) in the eastern North Atlantic and (bottom) from the eastern North Pacific [*Sutton and Barstow.,*1990] Vertical components are labelled "Z"; horizontal components are labelled "H" or "E" and "N".
The thin lines on the top panels show the high and low noise continental site models.

Borehole and subsurface installations at long period

There are (at least) **four important noise sources** at periods longer than 10s that affect detection of seismic phases:

- The *primary frequency microseism peak from 0.06 to 0.1 Hz* determines the *vertical component noise level in the band from 10- to 20-s period*. This energy propagates as fundamental mode Rayleigh waves, and no improvement in S/N-ratio is expected in this band unless the seismometer is installed at very great depth (>10km).
- *Currents* control *horizontal component noise at frequencies below 0.1 Hz* at sites on the seafloor. Although one of the most important reasons cited for installing long-period seismometers in boreholes is the avoidance of noise due to bottom current [*Purdy and Dziewonski*, 1988], it is likely that shallow burial (2 m) will almost always be adequate to avoid this source of noise at marine sites. Only at some unusual locations will deep ocean currents produce turbulent boundary layers comparable in magnitude to wind-driven pressure fluctuations [*Webb*, 1988]. At these sites it might be useful to bury sensors as deep as 60m.
- *Biological noise* can be a problem at *all frequencies*, as sea life bumps into sensitive seismometers. Shallow burial should put the seismometers out of reach of animals.
- ***Infragravity waves* provide the strongest incentive to consider deep borehole installations for long-period sensors, although the achievable improvement in S/N-ratio is small** (± 10 dB less noisy at 400m below the seafloor). The smaller infragravity wave signal at depth is due to a combination of a larger rigidity for rock compared to sediments and the evanescent decay of the signal with depth. Reducing the noise level in the band below 100s will provide better records of surface waves.

But long-period noise levels from BB seismometers installed in seafloor boreholes have been a disappointment, probably because of the flow in the borehole driven by the hydrothermal gradient.

✔ **Microseism band** (0.1 – 5 Hz)

Energy is coupled from ocean waves into acoustic waves and then into elastic waves below the seabed to form the microseism peak in ocean floor spectra. **Microseisms** are ground motions associated with this main spectral peak at 0.2 Hz (the *double frequency peak*). They propagate mostly as fundamental mode Rayleigh waves. The *single frequency peak* is a much smaller peak located between 0.05 – 0.1 Hz. The two peaks are created by separate mechanisms. Microseism amplitudes on *horizontal components are comparable to the vertical component* near 0.2 Hz and usually slightly higher at frequencies near 1 Hz.

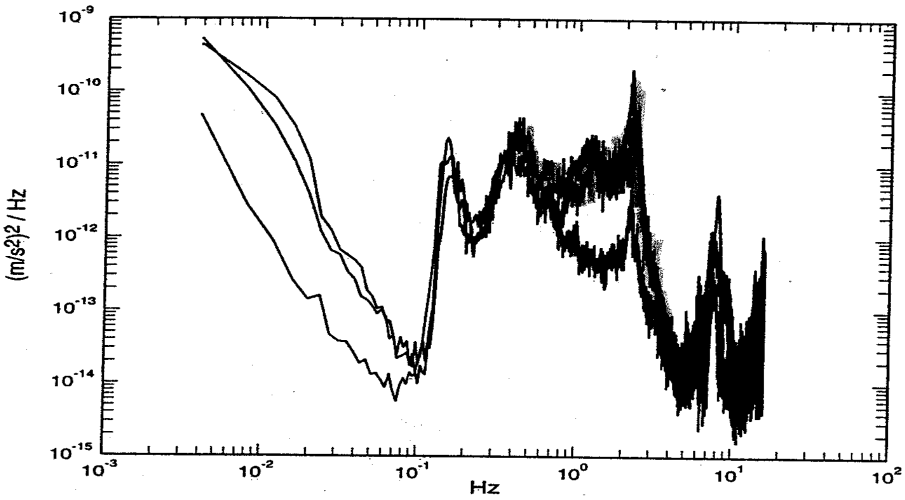


Fig. A20: Horizontal and vertical component acceleration noise spectra from a 1-Hz seismometer deployed on the East Pacific Rise. Long-period vertical noise level is determined by instrument noise. Horizontal noise is due to tilt and is quite variable over time. Large shear mode resonances or coupling resonances are very apparent at this site above 3 Hz.

A series of peaks is usually seen in the horizontal component spectra above 1 Hz. These peaks are probably due to short-wavelength shear modes (Stoneley or Scholte waves), but instrument coupling resonances may also contribute. Energetic microseisms driven by local ocean waves are present in ocean floor spectra at frequencies as high as 5 Hz. Usually, three peaks are evident between 0.05 and 1 Hz in spectra from sites in the Pacific.

- The lowest frequency (*single frequency*) peak between 0.05 and 0.1 Hz often appears only as a shoulder to the main peak at 0.2 Hz.
- The main peak (*double frequency)* between 0.13 and 0.2 Hz is more variable and is associated with more local storms in the Pacific.
- A third peak at higher frequencies is usually related to the local wind wave field [*Webb*, 1992].

The microseism noise level near 1 Hz determines whether short-period compressional body waves can be seen from a teleseismic earthquake.

Temporal and geographical variation of microseisms

Geographical and temporal variations in vertical component seafloor noise are probably larger than any possible improvement in S/N-ratio that can be obtained by better installation techniques. Choosing better sites may offer bigger gains in S/N-ratio than drilling deeper boreholes.

The high seismic noise level at the seafloor near 1 Hz is caused by the interaction of ocean waves at the surface of the ocean. Pairs of ocean waves travelling in opposite directions can interact to couple energy into elastic waves. The frequency is doubled from the ocean wave spectrum into the microseism excitation spectrum. In the Pacific, the microseism spectrum usually has several peaks. The excitation could be calculated from the frequency and directional spectrum of the ocean wave field. The microseism spectrum saturates because of a balance between excitation by ocean waves and attenuation within the Earth. For the purpose of detecting short-period teleseismic body waves, one should seek sites with calm winds.

Because the ocean floor seismic noise spectrum depends on the wind, it varies with location and season.

- *Long-period* microseisms are always present because they propagate long distances in the form of fundamental mode Rayleigh waves from large storms.
- At periods < 5 s microseisms at a site are "regional" , shorter-period waves dissipate more quickly and travel shorter distances

The seafloor is usually noisy at 1 Hz because these microseisms are caused by 0.5 Hz (2 s period) ocean waves, and even a moderate breeze will quickly produce these.

- *Short-period* microseism noise level falls significantly only during periods of very light winds.
- The nearly continuous ice coverage of the Arctic Ocean in winter prevents the excitation of ocean waves, so that the Arctic seafloor pressure noise levels in the microseism band are lower than the levels at any site yet measured in the Atlantic and Pacific Oceans [*Webb and Schultz,* 1992].

In the compilation of seafloor vertical displacement spectra from various sites (Figure A21), the noise level at 1 Hz at the noisiest site was 40 dB above the quietest site. The Pacific has generally higher noise levels than the Indian Ocean or the North Atlantic.

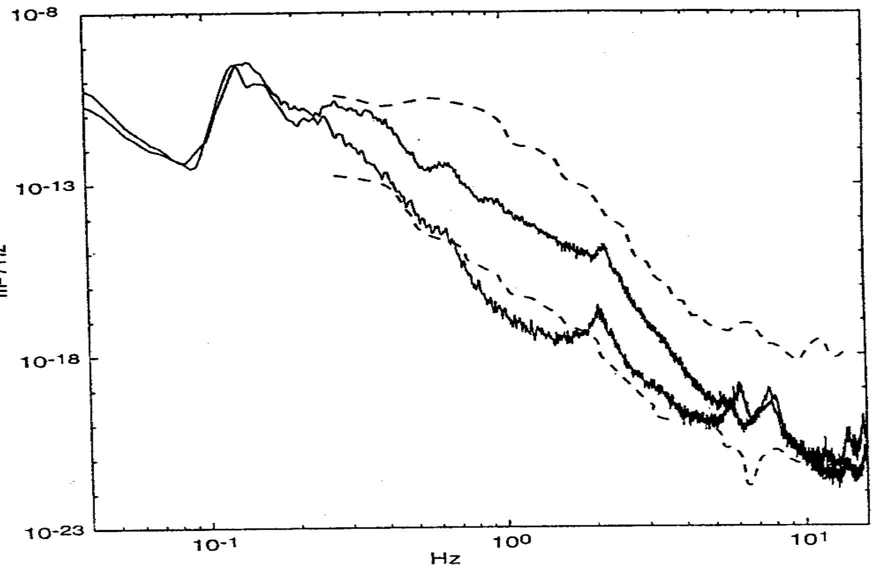


Fig. A21: Noisy and quiet days from the East Pacific Rise plotted as a displacement spectra along with the noisiest and quietest spectra from a large series of OBS experiments from *Hedlin and Orcutt* [1989].

Climatology of short-period microseisms

Noise levels in the microseism peak at 1 Hz are determined by the ocean wave spectrum at 0.5 Hz in an area local to the site. Winds above 5 m/s are likely to generate 2s ocean waves and therefore 1s microseisms. Noise levels near 1 Hz increase by 20 dB from calm to windy conditions and saturate at wind speeds exceeding 10 m/s [*McCreery et al.,*1993].

Higher-frequency microseisms need less wind to become saturated, noise levels at 2 Hz increase by about 12 dB over calm conditions and reach saturation at about 7 m/s wind speed.

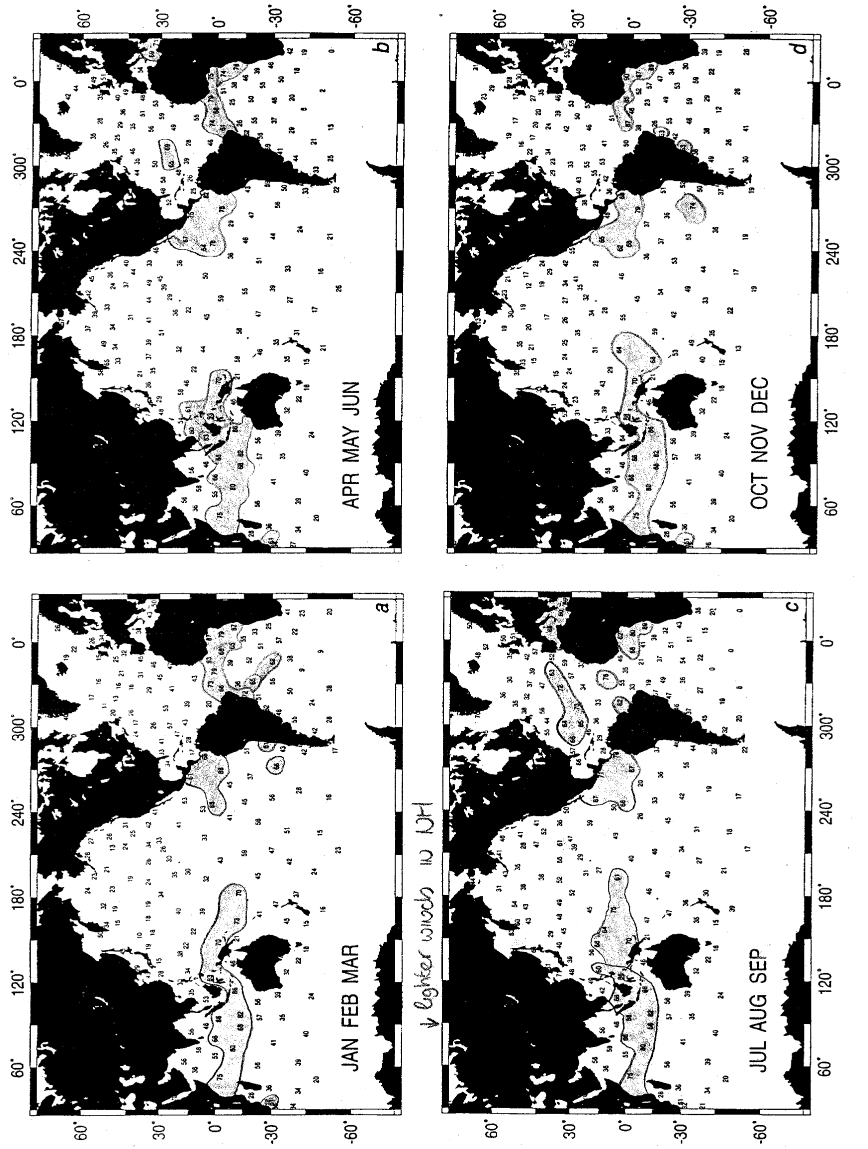


Fig. A22: The numbers indicate the percentage of time in a year when windspeeds are less than 5 m/s. In the shaded areas the winds are less than 5 m/s more than 60 % of the time (at 10-m elevation above the sea surface).

The shaded areas in figure A22 are likely to provide low seafloor noise ratios for short-period arrivals from teleseismic earthquakes of moderate amplitude (m_b > 5.5). In general, the Atlantic & Indian oceans are less windy than most of the Pacific. There is a strong seasonal cycle to the probability of light winds. There is a need for systematic investigation of short-period noise levels with the location on the ocean floor. The above map was designed to provide a guide for planning permanent stations and for planning temporary OBS deployments.

In *the Pacific* it is unusual to detect short-period arrivals from teleseismic earthquakes because so much of the Pacific is windy and therefore noisy [*Blackman et al*.,1995]. The *central Atlantic* near 35°N, 35°W, is often calm (72%) during early summer where short-period arrivals from distant earthquakes in South America with Ms < 5.6 were recorded [*Forsyth,* 1996]. The *Indian Ocean* near 25°S, 70°E, is usually calm (80%) during the month of august and there too, short-period arrivals were seen from six teleseismic earthquakes during a 3-week period [*Sato et al*., 1996].

✔ **VLF acoustics band** (5 – 50 Hz)

5 – 10 Hz : associated with wave breaking [*McCreery et al*., 1993]
10 –15 Hz : mostly controlled by man-made sources, primarily shipping [*Wenz*, 1962] (but large ships can produce energetic spectral lines at frequencies as low as 1 Hz).
> 15 Hz : whales and other marine mammals produce significant sound levels

The VLF band is the traditional band of marine seismology encompassing signals from microearthquakes and from the airgun and explosive sources used in refraction and reflection studies. The only significant teleseismic phases in the VLF band are the oceanic P_n and S_n phases (also called Po and So):

- which travel within the high-Q oceanic lithosphere
- arrivals can have energy at frequencies as high as 15 –20 Hz at distances greater then 30° [*Walker et al.,*1983; *Butler et al*., 1987].
- Efficient propagation requires a low-attenuation oceanic lithosphere
- On older oceanic crust they allow the detection of moderate size earthquakes at quite large distances

Coupling problem

The fidelity of short-period observations using ocean bottom seismometers is limited by the sensor coupling to a sedimented or rubble-strewn seafloor. The typical seafloor installation of an OBS consists of dropping the instrument into the water and letting it fall on whatever patch of seafloor is below. Most of the seafloor is covered with a layer of weakly consolidated sediment, with shear velocities as low as 25 m/s [*Schreiner and Dorman*, 1990]. Therefore it is almost impossible to obtain a high coupling frequency for horizontal motion, *Trehu and Solomon* [1981] report a horizontal coupling frequency of 6 - 10 Hz with a vertical coupling frequency of 22 Hz on soft sediment for the MIT OBS's.

For most seismic phases there is a differential motion between the water and the seafloor, and the net motion of the sensor is a combination of the seafloor motion and the motion of the water above the seafloor. Horizontal motions of the seafloor must invariably tilt the seafloor sensor package coupling into both horizontal and vertical motions [*Duennebier and Sutton*, 1995] unless the sensor is installed below the seafloor into the sediment. This can be done either by jetting the sensor into the bottom with a flow of water using a remote-operated vehicle (ROV) [*Yamamoto et al.,* 1989], installing it in a caisson with a ROV into which the sensors are installed [*Duennebier et al.,* 1991], or driving it with a hydraulically driven screw that pulls the sensor package into the mud to a depth of as much as 10m. The increasing shear strength at depth will also improve coupling.

Another solution is to install the sensors into boreholes, but it can be difficult to couple the sensor to the uneven wall of the borehole, and the casing of the borehole may not be tightly fixed to the surrounding rock. Boreholes could be filled with sand or the sensors should be grouted into the borehole with cement to avoid noise from flow in the borehole. Noise coupled into the borehole by motions of the massive re-entry cone at the top of the borehole must also be considered a possible source of seismic noise in ocean floor boreholes.

In practice, the fidelity of sensor measurements at high frequency has been of secondary importance. The primary interest has been to detect the shear and compressional wave arrivals clearly. At periods longer then 1s, coupling seldom appears to be a problem on either sediments or rock.

Detection limits for teleseismic short-period body waves and the microseism band (0.1 – 5 Hz)

The acceleration amplitude spectrum of a teleseismic, P- wave arrival from a large earthquake is peaked toward frequencies near 1 Hz. The dominant frequency of an arrival depends primarily on the *attenuation along the ray path*, which removes the high frequency components for which the noise background is lower. Attenuation is higher

- for earthquakes at teleseismic distances than at regional distances (<10°). At regional distances body waves from small-magnitude (Mb < 3) earthquakes are readily detected between 5 and 10 Hz because attenuation and geometrical spreading are small at these short distances. At teleseismic distances, the detection limit for short-period body waves will fall between Mb = 5.5 (deep earthquake & quiet ocean surface) and Mb > 7.5 at noisy seafloor sites like OSN-1.
- for shallow events than for deep sources because rays travel only once through the attenuative upper mantle.
- in tectonically active regions than in shield areas.
- in oceanic regions than in continental regions and probably much higher under oceanic ridges.
- On younger oceanic crust because the high-Q lithosphere is thicker.

It is expected that an arrival will only be detected if

- the predicted amplitude substantially exceeds the noise
- the peak P-wave amplitude is about 20 times larger than the rms noise [*Aki*., 1976]

For the moment it is assumed detection will occur when the signal model curves exceed the noise curves by a factor of 6 (16 dB).

Horizontal noise levels near 1 Hz tend to be about *10 dB higher than vertical noise levels* (see Figure A20), so that teleseismic, short-period P-waves will only be seen at the quietest sites. Shear waves are usually about a factor of 3 larger and should be seen on the horizontal components whenever the P-wave is visible on the vertical component.

A study of records from 11 OBS experiments (mostly at ridge crests) confirms the mostly pessimistic assessment of detection limits in the Pacific basin [*Blackman et al.,* 1995]. In contrast, short-period arrivals have been well recorded during OBS experiments in the Atlantic and Indian Oceans. **At most seafloor sites, short-period P waves from distant (>30°) earthquakes will be detected only from the very largest earthquakes because of high noise levels and because the more easily detected high-frequency components are removed by attenuation. High noise levels near 1 Hz may raise detection limits for short-period, teleseismic (ranges > 30°) arrivals above m_b =7.5, m_b =4.5 on land (see appendix A for scale definitions).** This high detection threshold is a problem for Earth tomography.

Ocean floor currents and short-period noise

Currents can sweep some parts (along western edge of basins, around bathymetric features, at the top of mid-ocean ridges] of the deep ocean floor with speeds as high as 75cm/s [*Rhines,*

1977; *Gross et al.,* 1986]. In most other places currents are weak, rarely exceeding 10 cm/s and less than 5 cm/s at many sites. But currents as small as 10 cm/s can produce *significant short-period noise on both vertical and horizontal components* of poorly designed (narrow and tall) OBS's [*Duennebier et al.,* 1981].
Currents stronger than 10 cm/s increase short-period noise levels even on low-profile instruments, noise levels rise as the current velocity to the fourth power [from *Trehu*, 1985b]. To minimize current induced noise, shallow burial of the seismometer is most effective because sensors placed in boreholes can be affected by water circulation inside the borehole driven by a geothermal gradient.

Short-period noise in boreholes into the seabed

Seismometers are installed in boreholes on land to escape noise associated with the local deformation of the ground under atmospheric pressure fluctuations, ocean currents are a similar source of noise at the seabed. There have been experiments investigating noise levels in boreholes in the seabed, primary goals were:

- To improve the S/N-ratio for short-period signals
- To improve sensor coupling to the seabed

Important noise sources at periods shorter than 10s that affect detection of seismic phases:

- ***Microseism energy scattered by bathymetry and by topography*** on the sediment-basement interface into short-wavelength shear modes. Some improvement is expected by burial at shallow depths
- *Microseisms.* This short-period noise decreases with depth and as a function of frequency [*Stephen et al*.,1994; *Bradlley et al*., 1997]. On the horizontal components the decrease in noise levels is larger. Most of the improvement occurs in the first 10 m for frequencies above 1.5 Hz, but below 1 Hz the sensor at 10 m depth was only a few decibels quieter than the seafloor sensor. The depth dependence at 0.3 Hz was consistent with propagation as fundamental mode Rayleigh waves with little difference between seafloor and borehole sensors. The improvements in S/N-ratio are obvious in the different structure of the borehole and OBS noise spectra (Figure A23).
- *Currents* can produce significant short-period noise on both vertical and horizontal components of poorly designed OBS's Shallow burial (2 m) will almost always be adequate to avoid this source of noise
at marine sites.
- *Biological noise* can be a problem at *all frequencies*, as sea life bumps into sensitive seismometers. Shallow burial should put the seismometers out of reach of animals.

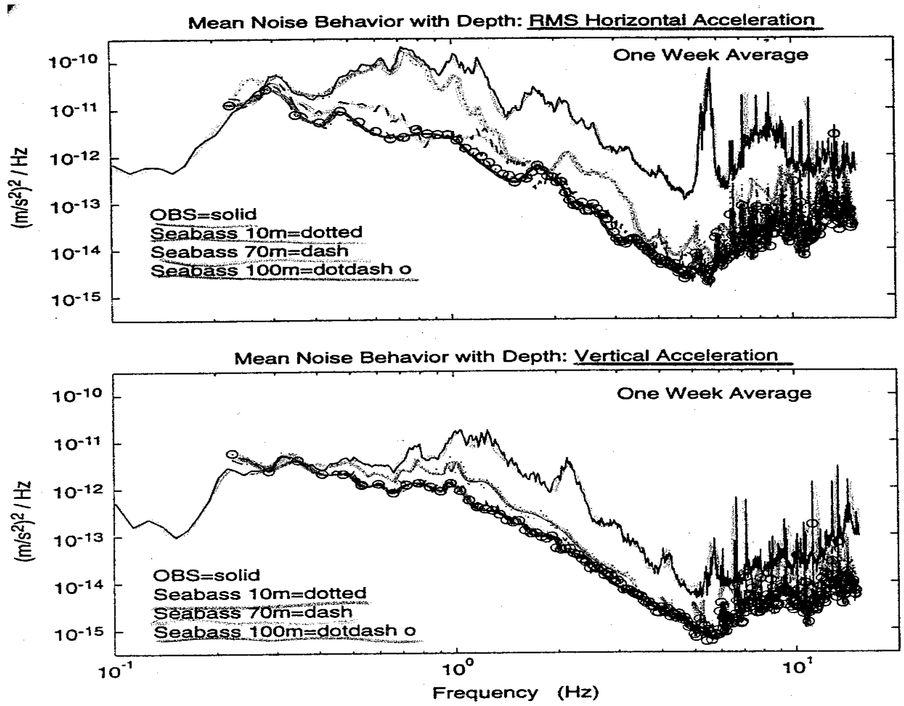


Fig. A23: (top) Horizontal and (bottom) vertical noise spectrum from a North Atlantic site on the seafloor and at 10, 70, and 100 m below the ocean floor in the Deep-Sea Drilling Project borehole 534B [from *Bradley et al.,* 1997]

The deep spectra closely follow a power law dependence on frequency with a slope of 80 dB/decade [*Aider et al.,* 1984]. In contrast, surface spectra show significant peaks in the band around a few Hertz associated with the Stoneley or shear modes. The horizontal and vertical spectra at depth differ by only a few decibels consistent with propagation of the microseisms as Rayleigh waves.

Vertical noise levels are lower within the basement rocks, but signal levels in boreholes are also smaller (because of changing impedance). The resulting improvement in S/N-ratio is near zero at 0.2 Hz and increases to more than 10 dB at a few hertz. For the horizontal components the improvement is much larger and occurs mostly within the upper 10m.

Borehole installation can provide a small but significant improvement in S/N-ratio for short-period arrivals compared with seafloor installations. The improvement is less than 10 dB for vertical component measurements and 10 – 15 dB for horizontal sensors corresponding to 1 – 1.5 units of magnitude in detection threshold. One measurement of depth dependence suggests that all improvement of S/N-ratio occurs in the first 50 m, therefore shallow burial of sensors is sufficient to attenuate short-period flow noise to negligible levels.

The need to improve S/N-ratio for short-period teleseismic phases provides the most compelling reason for installation of seismometers into deep boreholes into the seabed. The depth of burial is important because it greatly affects the cost of an installation. A deep borehole into rock can only be drilled using the Ocean Drilling Program (ODP) ship, which costs about $60.000 per day to operate and is heavily subscribed for other work.

Part I

Studying tilt-noise on oceanbottom seismometers

Introduction

The coherence between vertical seismic and pressure measurements indicates the quality of low-frequency vertical component data. Low coherence in the frequency band 0.002-0.04 Hz suggests that other noise sources, such as tilting due to ocean currents, dominate the "background" noise level. Low frequency (< 0.1 Hz) vertical component noise on seafloor seismometers can be reduced by ± 25 dB by subtracting the coherent signals derived from (1) horizontal seismic observations associated with *tilt noise*, and (2) pressure measurements related *to infragravity waves*. The reduction is largest for the poorest stations: sites with soft sediments, high currents, shallow water or a poorly developed seismometer. Low-frequency background vertical seismic spectra measured on a seafloor seismometer levelled to within 1 x 10^{-4} radians (0.006 degrees) are up to 20 dB quieter than on a nearby seismometer levelled to within 3 x 10^{-3} radians (0.2 degrees). The noise on the less precisely levelled seismometer increases with decreasing frequency and is correlated with ocean tides, indicating that it is caused by tilting due to seafloor currents flowing across the instrument. At low frequencies, this tilting generates a seismic signal by changing the gravitational attraction on the geophones as they rotate with respect to the earth's gravitational field. The effect is much stronger on the horizontal components than on the vertical, allowing significant reduction in vertical component noise by subtracting the coherent horizontal component noise. This technique reduces the low-frequency vertical noise on the less-precisely levelled seismometer to below the noise level on the precisely levelled seismometer. The same technique can also be used to remove "background" noise due to the seafloor pressure field (up to 25 dB noise reduction near 0.02 Hz) and possibly due to other parameters such as temperature variations. These are the findings of *Wayne C. Crawford and Spahr C. Webb* in their paper "*Identifying and removing tilt noise from low-frequency (<0.1 Hz) seafloor vertical seismic data*". High horizontal noise levels are also present in the spectra of OSN and H2O sites. I've attempted to apply the theory of Crawford and Webb to these new broadband seismometer data, in order to improve the Signal to noise-ratio at the OSN site. Because this paper was the basis of my research, a short summary of their findings is given in the next chapter.

Scientific background

I : THE EXPERIMENT

Two pressure-acceleration sensors were deployed in 900m deep water at 31°24'N, 118°42'W, at the outer edge of the California Continental borderlands. The sediments are about 1 km thick, and the seafloor is relatively flat. The instruments, deployed 1 km apart, collected data from September 10 to September 24 1998.

One instrument carries:

- A differential pressure gauge [*Cox et al*., 1984]
- A seismometer: Lacoste-Romberg gravimeter [*Lacoste*, 1967]
  - One vertical channel
  - Long period [*Agnew et al.,* 1976]
  - Sample rate = 2 samples/sec → Nyquist frequency = 1 Hz
  - Levelled using motorized gimbals to within 5 x 10^{-5} radians of a laboratory-determined value
  - The centre value of uncertainty is approximately 1 x 10^{-4}

The other instrument carries:

- A differential pressure gauge
- A Streckeisen STS-2 seismometer
  - Three channels, X and Y (horizontal) and Z (vertical)
  - Broadband
  - Small rate = 2 samples/sec → Nyquist frequency = 1 Hz
  - Levelled using motorized gimbals to within 5 x 10^{-3} radians of its centre value
  - The electronics correction is accurate to within 1 x 10^{-2} radians

II : SEAFLOOR SEISMIC SPECTRA

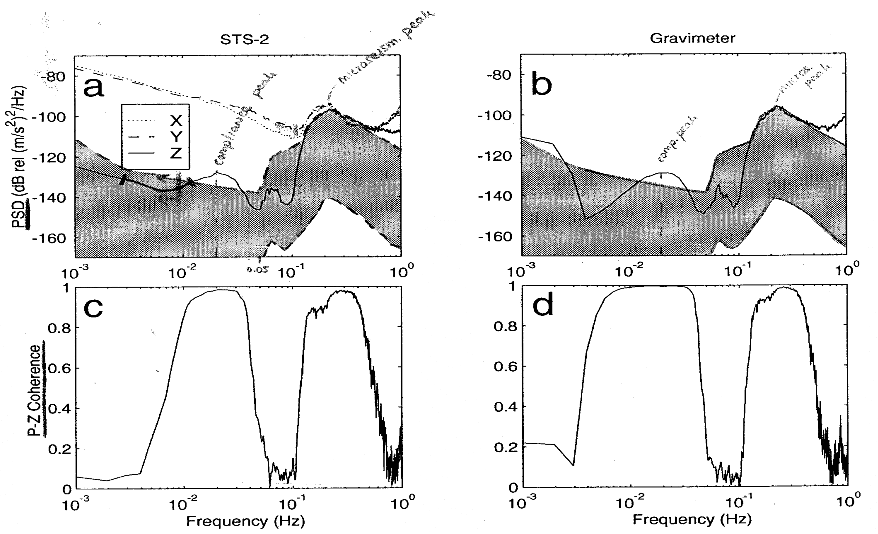


Fig. T1: "Background" seismic autospectral densities and pressure-vertical coherence from two seafloor seismometer/pressure gauge packages deployed 1 km apart on the 900m deep seafloor. The shaded shows the bounds of seismic noise observed on land stations [*Peterson,* 1993].

The spectra of both instruments are calculated from Finite Fourier Transforms [*FFT, Bendat and Piersol*, 1986] of data windows visually inspected to avoid seismic events.
The seafloor seismic spectrum includes:
- A microseism peak at frequencies > 0.1 Hz
- A "compliance" peak centred around 0.02 Hz
- A red noise spectrum at lower frequencies

At frequencies below 0.05 Hz, the STS-2 is noisier than the gravimeter. This is caused by "leakage" of horizontal noise to the vertical component due to the geophone being tilted from the vertical. At frequencies below 0.1 Hz, horizontal channels are more than 45 dB noisier than vertical channels.

The "unavoidable" noise source: seafloor compliance

Seafloor "compliance" means the seafloor deformation under pressure forcing from linear surface gravity waves.
- It is only significant if the *wavelength* of the surface gravity wave is longer than the waterdepth (H). For a waterdepth of 4000m it will give noise at frequencies below 0.02 Hz, so in deeper water at lower frequencies.
- The *amplitude* depends mostly on the strength of the pressure signal, which is approximately constant in the Pacific Ocean, and on the shear modulus of the underlying basement.
- It can be used to *study the shear modulus / shear velocity structure* of the sediments and crust [*Crawford et al.,* 1999] or it can be *removed to allow easier detection* of other seismic signals [*Webb and Crawford*, 1999]
- It is *2-4 times smaller on the horizontal channel* as it is on the vertical, but this is usually far below the horizontal background spectral levels.
- It can be used to evaluate the vertical seismic data quality. If there are no other significant noise sources, the pressure acceleration coherence amplitude will be 1. A decrease in coherence indicates that there is another noise source; coherence less than 0.5 indicates the other noise source is stronger than the compliance signal.

The pressure-gravimeter coherence has larger amplitude and spans a larger frequency range than the STS-2 pressure-vertical coherence, indicating that the STS-2 vertical senses more non-compliance noise than the gravimeter. The smaller amplitude and the higher cut-off frequency of the STS-2 pressure-vertical coherence indicates an additional low-frequency noise source that increases with decreasing frequency.

The "extraneous" noise source: seafloor currents

Several lines of evidence indicate that spectral noise levels below 0.01Hz (V) and below 0.1Hz (H) are dominated tilting of the instrument due to currents:
- Noise levels vary in sync with ocean tides
- The slope of the low-frequency noise is the same as that observed for seafloor currents
- The horizontal and vertical spectra have similar slopes and are coherent, with much larger noise levels on the horizontals
- The noise is not seen on the precisely levelled gravimeter

The STS-2 background noise below 0.1Hz fluctuates with the tides. The noise is maximum when the first derivative of the local ocean tides is maximum, and minimum when the first derivative of the local ocean tides is minimum (Figure T2).

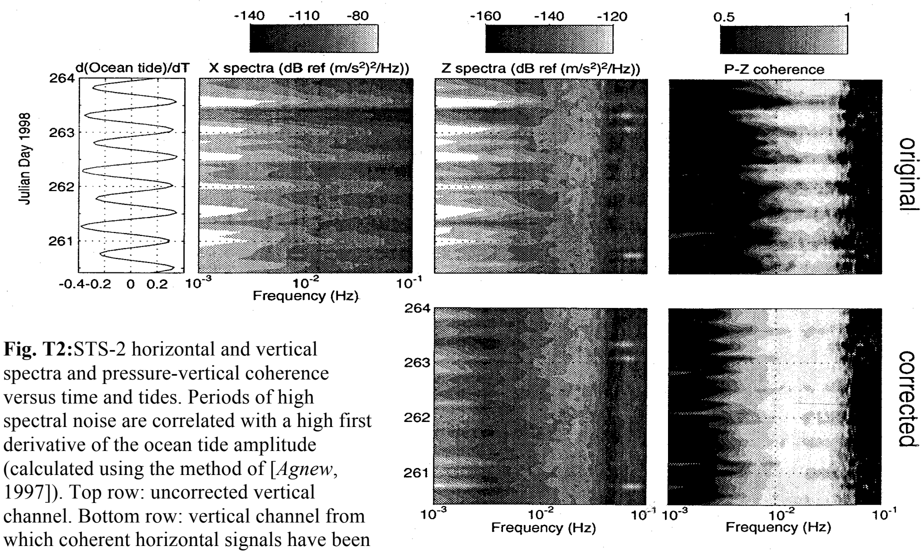


Fig. T2:STS-2 horizontal and vertical spectra and pressure-vertical coherence versus time and tides. Periods of high spectral noise are correlated with a high first derivative of the ocean tide amplitude (calculated using the method of [*Agnew*, 1997]). Top row: uncorrected vertical channel. Bottom row: vertical channel from which coherent horizontal signals have been removed.

Seafloor currents are primarily tidally driven although near-intertidal motions can contribute significantly [*Thomson et al.,* 1990; *Brink,* 1995]. Current-induced tilt noise is caused by seafloor currents flowing past the instrument and in eddies spun off the back of the instrument [*Webb*, 1998; *Duennebier and Sutton*, 1995]. The effect of seismometer tilting on the Acceleration signal can be modelled by assuming that the geophones at rest are rotated by an angle θ from the vertical and are offset from the instrument centre of rotation by a distance L and an angle ϕ_0.

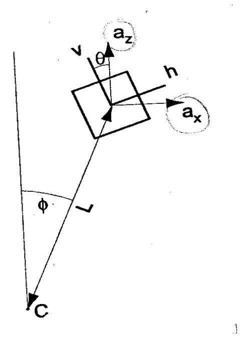


Fig. T3: Schematic drawing of the geometry used to calculate the effect of tilt on the geophone signal. The square represents the geophone and C is the centre of rotation

Current forcing rotates the geophones around the centre of mass. This rotation creates acceleration signals on the geophone by two processes (1) a "**displacement**" term that is the second derivative of the geophone position and dominates at higher frequencies (2) a "**rotation**" term that comes from the change in the gravitational acceleration on the geophones and dominates at lower frequencies (Figure T3). The rotational term is generally much larger on the horizontal channels than on the vertical, so low-frequency tilt noise is much larger on the horizontal channels than on the vertical, furthermore it is very small when the instrument is perfectly levelled but increases greatly when the geophone is even slightly off level.

The displacement of the geophone from the instrument centre of mass, which plays a large role at higher frequencies, has no effect at frequencies below 0.1Hz. A simple rotation of the geophones to the vertical should reduce the vertical tilt noise level to near or below the continental noise levels [*Peterson*, 1993].

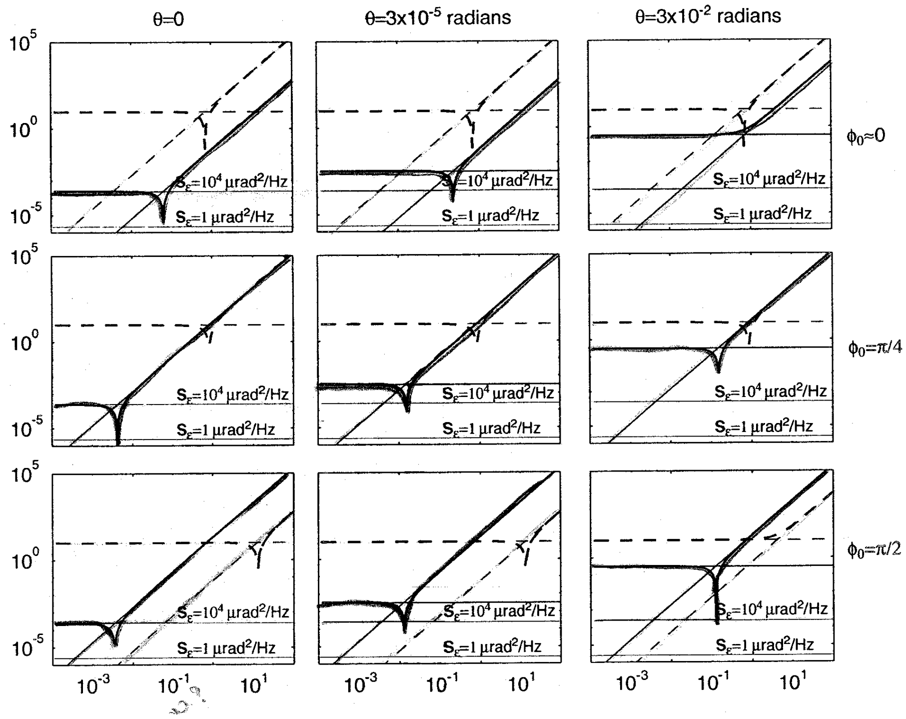


Fig. T4: Tilt-acceleration transfer functions, assuming the geophone is 0.5 meters from the centre of rotation Thick lines show the overall transfer function, while thin lines show the "rotation" (zero slope) and "displacement" (positive slope) components. The only important non-linear term is the vertical acceleration due to rotation, which is significant if the geophones are perfectly levelled (θ=0). The thinnest horizontal lines show the transfer functions due to this non-linear term, for constant tilt spectra $S_t(f)$ = 1 and 10^4 μradian^2/Hz. Each row has the same angle between the geophone and the centre of rotation (ϕ_0) and each column has the same geophone tilt from the vertical (θ).

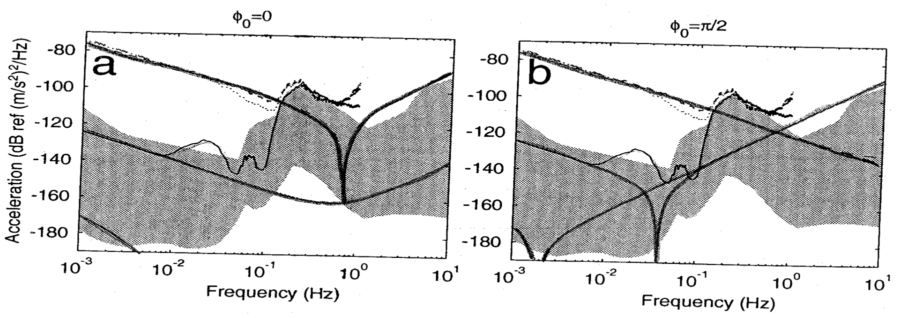


Fig. T5: Modelled acceleration signals (grey lines) versus seafloor STS-2 spectra (black lines). Dashed lines show the horizontal components, solid lines show the vertical component. Assumed tilt spectrum $\phi_t(f) = 1.3x10^{-2} f^{1.5}$ μradian^2/Hz, L=0.5m. The upper solid grey line is the modelled vertical acceleration for a geophone tilt of $3x10^{-3}$ radians, and the lower solid grey line is the modelled vertical acceleration for a perfectly levelled geophone. A) Assuming the geophone is directly above the centre of rotation (ϕ_0=0). B) Assuming the geophone is to the side of the centre of rotation ($\phi_0=\pi/2$).

The coherence between the three seismometer components helps to illuminate the different background noise sources:
Frequencies > 0.1Hz → seismic channels are all dominated by microseism energy, so they are all partly coherent with one another
Frequencies < 0.1Hz → horizontals are dominated by tilt noise, horizontal-vertical coherence comes from tilt noise on the vertical. This coherence is high except over peaks in the vertical seismic energy (the infragravity wave peak at 0.01-0.04Hz and the small peak at 0.07).

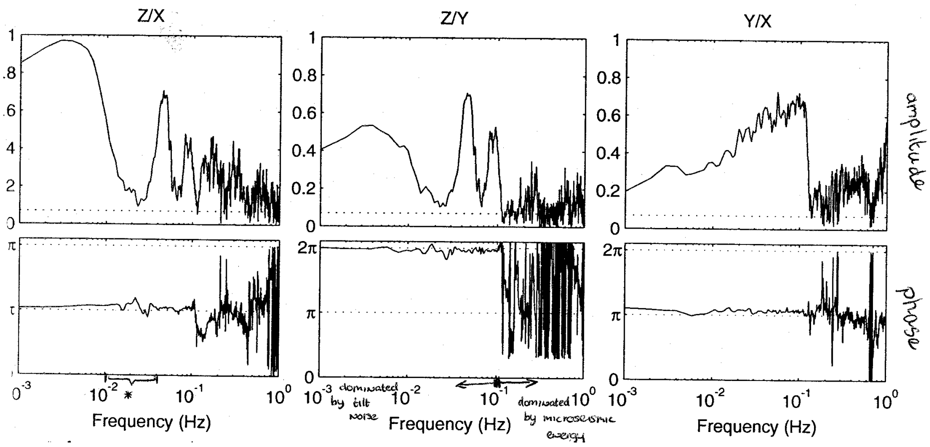


Fig. T6: Coherence between STS-2 seismometer channels. Upper plots show amplitude, lower plots show phase. Z is the vertical channel, X an Y are the horizontal channels. The dotted line in the amplitude plots marks the 95% significance level.

III : REMOVING TITLT NOISE FROM THE VERTICAL CHANNEL

To remove the low-frequency tilt noise from the vertical channel, it is assumed that
- The horizontal channels are completely controlled by tilt below 0.1 Hz
- The infragravity signal is only on the pressure and vertical channels

The tilt-generated noise is removed from the vertical channel by
- Calculating the transfer function* between the horizontal and vertical channels
- Subtracting the coherent horizontal energy from the vertical channel

This technique is inferior to rotation-back –to-vertical if non-linear terms are important (other horizontal/vertical noise coupling not accounted for by the tilt model) may be removed, but it is preferred because of it's more general applicability. Non-linear terms are generally insignificant compared to instrument noise levels. This method does not work above 0.1 Hz because the seismometer channels are all dominated by the "microseism" signal.

To calculate the *frequency domain transfer functions* between the different channels [*Bendat and Piersol*, 1986]:
1) Estimate the **one-sided autospectral density** functions G_{ss} and G_{rr} from the Fourier transforms* of windowed sections of data for variables *s* and *r*

$$G_{ss}\left(f_j\right) = \frac{2}{n_d N \Delta t} \sum_{i=1}^{n_d} \left|S_i\left(f_j\right)\right|^2, \qquad j = 0,1,\ldots,\frac{N}{2} \tag{1}$$

* Appendix B

$$G_{rr}(f_j) = \frac{2}{n_d N \Delta t} \sum_{i=1}^{n_d} \left| R_i(f_j) \right|^2, \qquad j = 0,1,\ldots,\frac{N}{2} \tag{1}$$

2) Estimate the **one-sided cross-spectral density** function G_{rs} from the Fourier transforms* of windowed sections of data for variables *s* and *r*

$$G_{rs}(f_j) = \frac{2}{n_d N \Delta t} \sum_{i=1}^{n_d} R_i(f_j) S_i(f_j), \qquad j = 0,1,\ldots,\frac{N}{2} \tag{2}$$

3) Obtain the **coherence function $\gamma_{rs}(f)$**

$$\gamma_{rs}(f_j) = \frac{G_{rs}(f)}{\left[G_{rr}(f) G_{ss}(f)\right]^{1/2}} \tag{3}$$

s = the "source" channel (X, Y, or P)
r = the "response" channel (Z or Y)
n_d = the number of data windows
N = the length of each data window
Δt = the sampling interval

$S_i(f)$ = Fast Fourier transform* of data window *i* for the source channel
$R_i(f)$ = Fast Fourier transform* of data window *i* for the response channel

The **transfer faction** is:

$$A_{rs}(f) = \gamma_{rs}(f) \sqrt{\frac{G_{rr}(f)}{G_{ss}(f)}} \tag{4}$$

In the frequency domain the noise seen in component r is linearly related to the s component through the transfer function. The Fourier transform R' representing **R corrected for the noise in S** is then:

$$R'_i(f) = R_i(f) - A^*_{rs}(f) S_i(f) \tag{5}$$

The various transfer factions between Z,X,Y and P are calculated and the vertical component for noise is corrected in each of the other components in turn.
Before applying Eq. (4)

- Set A(f) = o for frequencies above 0.1 Hz to only subtract the part of the source channel that is noise on the vertical
- subtract out the coherent effect from all previous "sources", so the transfer function doesn't include information about signals already removed
- check if the transfer functions don't change with time

To confirm that all the coherent noise has been removed, recalculate the coherence between the three seismic channels. The corrected Z channel should now be incoherent with the X and Y channels for all frequencies below 0.1Hz.

<u>An example of noise removal in the frequency domain</u>

- the transfer functions between seismic channels are calculated for a relatively quiet time interval:

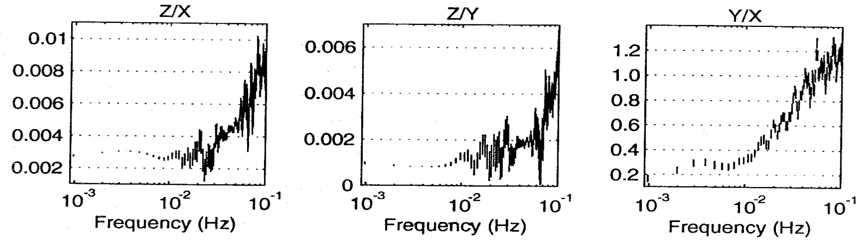


Fig. T7: Magnitude of transfer functions between seismic channels (the phase is the same as in the previous figure). The complex transfer function is used to subtract the effect of the horizontal channels on the vertical.

They are roughly constant below 0.01Hz (tilt noise dominates the vertical spectrum) and noisy and poorly resolved at higher frequencies (compliance signal dominates the vertical spectrum). Below 0.01Hz there is a 0.003 radian vertical component tilt from the true vertical in the x direction and a 0.001 radian tilt in the y direction. The effect of contamination of vertical component seismic waveforms by horizontal component seismic motions is small because the magnitudes of the horizontal to vertical transfer functions are small.

- The corrected vertical spectral noise levels are up to 20 dB lower than in the original spectrum:

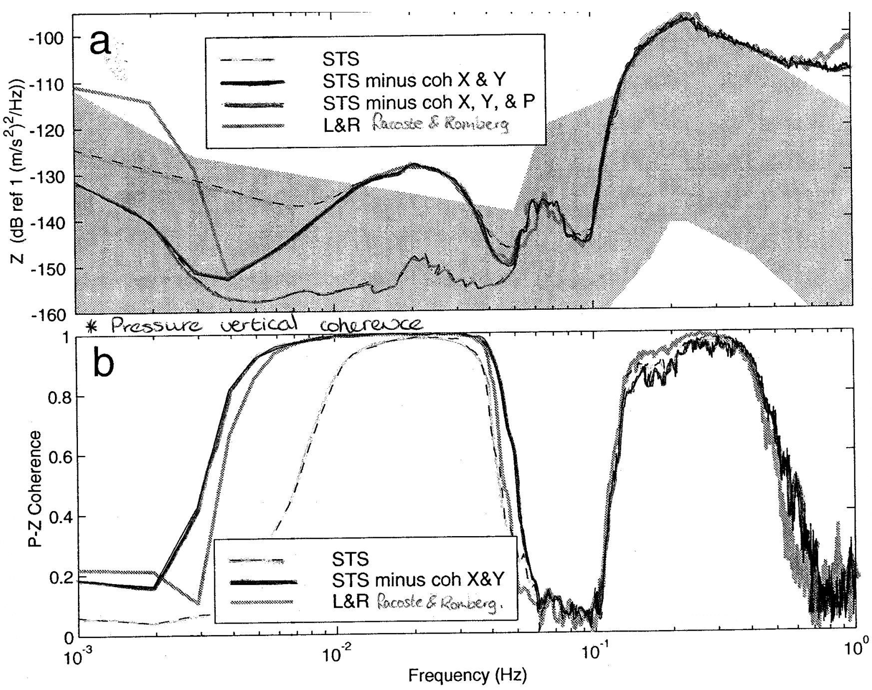


Fig. T8: Vertical spectra and vertical-pressure coherence before and after removing coherent noise from STS-2 horizontals. a) Autospectral densities. b) Pressure-vertical coherence

An example of noise removal in the time domain

The time domain correction is applied to a section of data containing arrivals from an earthquake. The tilt correction is significant, particularly for waveform modelling (Figure T9).

Other noise sources

The very low frequency gravimeter noise (Figure T1) probably comes from temperature fluctuations, since Lacoste-Romberg gravimeters are much more temperature sensitive than STS-2 seismometers.

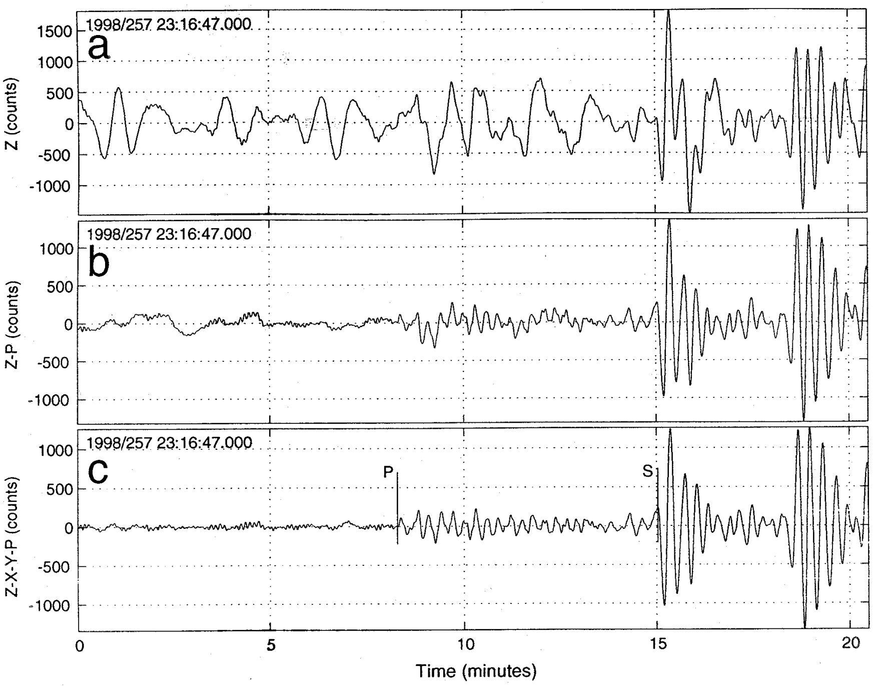


Fig. T9: STS-2 vertical seismic record of a magnitude 6.2 earthquake (Δ = 44.2°). All traces are band-pass filtered between 0.001 and 0.05 Hz. A) Original vertical trace. B) Vertical trace after subtracting coherent pressure signal. C) Vertical trace after subtracting coherent pressure and horizontal signals.

IV : DISCUSSION

This paper has concentrated on removing tilt-noise from the vertical component of a seismometer. The technique used can transform a marginal data set into a useful one, and is easily modified to account for other noise sources, even when the origin is not well understood.

Tilt-noise cannot be removed from the horizontal component because it isn't possible to construct a sufficiently accurate vertical reference (tilt sensor) that is also insensitive to horizontal acceleration.

In the measurements shown in this paper, the compliance noise is generally larger than the tilt-noise, but this is not generally the case. Because the seafloor seismometers in this case were designed to measure seafloor compliance [*Crawford et al.*, 1998], they were more

precisely levelled than typical seafloor seismometers. Low-frequency data at the OSN-1 [*Stephen et al.,* 1998] are dominated by tilt-noise.
The theoretical horizontal compliance signal is 2-4 times smaller than the vertical compliance peak but measured horizontal spectra, even in buried and borehole sensors, are almost always larger than vertical spectra [*Montagner et al*, 1994; *Collins et al*., 1998]. The horizontal compliance peak in the used data set would only be visible if the tilt spectrum is less than 10^{-6} $f^{-1.5}$ μradians ^{2}/Hz (dynamic tilts smaller than 0.03 μradians near 0.01Hz). Data from the H2O observatory show the first evidence for tilt levels this low.
Seafloor noise levels can probably be reduced further. The vertical seismic noise level after removing compliance and tilt effect is still well above the instrument noise floor.

Experimental method

✓ ***MAIN QUESTION:***

Can the quality of the vertical channel seismic data be improved by subtracting the coherent horizontal energy (which is assumed to represent tilt-noise) ???

Related questions and actions:

1. What does the long-period psd spectrum look like?
2. How important is the "unavoidable" noise source, seafloor compliance?
3. How important is the "extraneous" noise source, seafloor currents (resulting in tilt noise)?
4. What does the low-frequency coherence spectrum look like?
5. What does the transfer faction look like?
6. What is the difference between the vertical psd spectra before and after removing the coherent horizontal energy?

✓ ***DATA***

1 day (62) of OSN data:
/dat1/u/zeldenru/OSN/osn1_compressed/1998/096/19980960000.00.OSN1.BHZ
/dat1/u/zeldenru/OSN/osn1_compressed/1998/096/19980960000.00.OSN1.BH1
/dat1/u/zeldenru/OSN/osn1_compressed/1998/096/19980960000.00.OSN1.BH2

and 1 day (121) of H2O data:
/osn1/b3s2/OSN/H2O/work/h20_121_HH

✓ ***SOFTWARE***

I created MATLAB M-files* to do spectral analysis of the OSN and H2O data.
What is Matlab?
MATLAB is a computer program for people doing numerical computation, especially linear algebra (matrices). It began as a "MATrix LABoratory" program, intended to provide interactive access to the libraries Linpack and Eispack. It has since grown well beyond these libraries, to become a powerful tool for visualization, programming, research, engineering, and communication. Matlab's strengths include cutting-edge algorithms, enormous data handling abilities, and powerful programming tools.
The following steps were taken:

- t02s_tomgetspecdata_template1.m
- t03s_irenetest.m
- t04_puremfile.m
- t05_commentfile.m

These are a few different versions of the original M-file, which plots the Power Spectral Density of 2 channels (Z, rms of H+H1) of a borehole seismometer at the OSN site.
It also writes the calculated psd's to files in the 'spectra' directory. Explanations of M-commands are added. The goal of the original M-file was to build a script to loop through the whole data set in 131072 point chunks. For each chunk the code can compute a spectra. Then all of the spectra for the whole experiment should be plotted in a waterfall format (one each for vertical and horizontal).

---

* Appendix F

Tom wrote in the original M-file (**t02s_tomgetspecdata_template1.m**):
This script computes spectra continuously for the OSN borehole data set. The data is read directly from CSS (Steim compressed SEED) files sent by Frank in January '99. Spectra are computed for 131072pt blocks of data (1.82hrs). Within each block nfft=32768 with 75% overlap and Hanning tapered windows (13). fmin is 0.0006104Hz. Spectra are in absolute acceleration psd units. Read and compute matlab spectra (channels 1-3). Contiguous data starts at day 34. This code makes a loop through the first 13 chunks on day 62.

In **t03s_irenetest.m**, I changed the script so it would read data from .BHZ / .BH1 / .BH2 files (as above). These are the files in which the spectra of the three seismometer channels are saved. The used variables have the following meaning:

C1	= vertical component data / rms of horizontal components data		
C2 and C3	= horizontal spectral data	noverlapC	= window overlap
FC	= sampling frequency	Fft	= fast fourier transform
nfft	= length of fft in computing psd	NN	= length of the frequency array
dfagC	= detrending option	windowC	= width of the Hannin window
fC	= frequency array after Hanning window		
f	= frequency array after smoothing		
PxxC	= psd array of any given channel		
SAdb	= psd after TF54000 and smoothing		
TF54000	= M-file that computes the transfer function for the borehole system, it is defined by a magnitude vector, a phase vector and a frequency vector.		
Smooth_spec	= M-file that smoothes the spectra to approximately even spacing in log f based on the(column) vector of the frequencies (ff), the (column) matrix of one or more spectra (PP) and the scalar specifying number of frequencies / decade on output (NPD).		

The data are then manipulated in the following steps:
1) The data in the files are opened.
2) Binary data are read from the files.
3) The files are closed.
4) Matrices are created containing frequency and psd information of each seismometer channel.
5) A non-rectangular data window is applied to the sections prior to computing the power spectral density. This windowing reduces the effect of section dependence due to overlap, because the window is tapered to 0 on the edges. Also, a non-rectangular window diminishes the side-lobe interference or "spectral leakage" while increasing the width of spectral peaks. Overlap rates of about half the section length lower the variance of the estimate significantly. In this case a Hanning window with a width of 32768 samples is used and a sample overlap of 75%.
6) The psd is calculated. The psd function averages and scales the modified periodograms of detrended sections (the best linear fit is removed from the section) of a signal. Since the signal is real, psd returns only the frequencies from 0 trough the Nyquist frequency.
7) Amplitudes are adjusted.
8) Multiplication by delt to get correct root (Hz) behaviour.
9) Apply transfer function TF54000.
10) Smooth the spectra
11) The frequency array (f) and psd matrix (SAdb) are saved to be plotted in waterfall format
12) The layout of the plot is defined
13) The plot is printed

t04_puremfile.m is the same as t03 only without any comments, this was done so I could familiarise myself with the MATLAB script.

t05_commentfile.m is the same as t03 only now all the used MATLAB commands are added to the M-file, also for me to get more insight into the way MATLAB files are build up.

- **t06_timeplot.m**

This M-file plots the Signal in the Time Domain of rms H1+H2 of a borehole seismometer at the OSN site. The file is the same as t03 up until step (5), but instead of calculating the psd, the signal is plotted in the time-domain.

- **t07_threechan.m**
- t08s_threechan.m

This M-file plots the Power Spectral Density in the frequency domain of 3 channels (Z, H, H1) of a borehole seismometer at the OSN site. It also writes the calculated psd's to files in the 'spectra' directory. Explanations of M-commands are added in the t08s file.

- **t09_OSNcohere.m**

This M-file plots the Coherence between 3 channels (Z, H, H1) of a borehole seismometer at the OSN site. It also writes the calculated coherences to files in the 'spectra' directory. Plots the 3 coherence graphs on one page.
Cxy = cohere (x,y) finds the magnitude squared coherence between length *n* signal vectors *x* and *y* and the cross spectrum of *x* and *y*:

$$C_{xy}(f) = \frac{\left|P_{xy}(f)\right|^2}{P_{xx}(f)P_{yy}(f)}$$

x and *y* must be the same length.

- **t10_OSNcsd.m**
- **t10_H2Ocsd.m**

These M-files plot the CSD between 3 channels (Z, H1, H2) of the borehole seismometer at the OSN site and the buried seismometer at the H2O site. Plots the 3 csd graphs on one page.

Pxy = csd (x,y) estimates the cross spectral density of the length *n* sequences *x* and *y* using the Welch method* of spectral estimation.

- **t11_OSNpsd+coh.m**
- **t12_H2Opsd+coh.m**

These M-files calculate the Coherence between 3 channels (Z, H1, H2) of the borehole seismometer at the OSN site and the buried seismometer at the H2O site. It also calculates the psd for the same 3 channels. Plots the 3 coherence graphs and 1 psd graph on one page.
The following steps are taken:

1. Open files with data of channels z,h1,h2

* finds the discrete-time Fourier transform of the samples of the process and takes the magnitude squared of the result

2. Read data in chunks of 1.8 hours
3. Set parameters
4. Calculating coherence
5. Calculating psd
6. Plotting
7. Layout plotting

- **t13_OSNtransf.m**
- **t14s_H2Otransf.m**

These M-files calculate the Coherence between 3 channels (Z, H1, H2) of the borehole seismometer at the OSN site and the buried seismometer at the H2O site. From these it calculates the transferfunction between these three channels.
Plots the transferfunctions (magnitude/frequency) in 3 graphs on one page. The t14s_H2Otransf.m also writes the calculated transferfunctions to files in the 'spectra' directory. This following steps are taken:

1. Open files with data of channels z,h1,h2
2. Read data in chunks of 1.8 hours
3. Set parameters
4. Calculating coherence using the cohere MATLAB function
5. Calculating psd
6. Calculating the transfer functions
7. Plotting
8. Layout plotting

- **t15_easytransf.m**

This M-file opens files containing the transferfunctions of H2O data calculated in previous M-files. It plots them in either an amplitude or a phase graph.

- **t17s_OSNtransfa.m**
- **t16_H2Otransfa.m**

These M-files calculate the Coherence (using the method shown in the paper of Crawford & Webb) between 3 channels (Z, H1, H2) of the borehole seismometer at the OSN site and the buried seismometer at the H2O site. From these it calculates the transferfunction between these three channels. Plots the transferfunctions (angle/frequency) in 3 graphs on one page. The t17s_OSNtransfa.m also writes the calculated transferfunctions to files in the 'spectra' directory. This happens in the following steps:

1. Open files with data of channels z,h1,h2
2. Read data in chunks of 1.8 hours
3. Set parameters
4. Calculating csd
5. Calculating psd
6. Calculating the complex (root of) the coherence function from the psd's and csd's like it is done in the Crawford & Webb paper.
7. Calculating the transfer functions, depending on the square root of the coherence and the psd's of the two channels
8. Calculating the FFT of the time series of the three channels, necessary for calculating noise in ChanH2 corrected for coherent noise in ChanH1 and noise in ChanZ corrected for coherent noise in ChanH1

9. Converting back to time series by using inverse FFT
10. Calculating the csd and psd for Zc and H2c (the corrected vertical and horizontal channels) to calculate the corrected square root of coherence and with this the transfer function.
11. Calculating spectrum of ChanZc corrected for coherent noise in ChanH2c
12. Plotting
13. Layout plotting

- **t18_trytocorrectOSN.m**

First attempt to create M-file that will calculate and plot the corrected values for Z and H2 using transferfunctions and fft's. This file does not work, later version = 20_totalcorrect.m.
The following steps are taken:

1. Open files with data of channels z,h1,h2
2. Read data in chunks of 1.8 hours
3. Set parameters
4. Calculating csd
5. Calculating psd
6. Calculating the complex (root of) the coherence function from the psd's and csd's like it is done in the Crawford & Webb paper.
7. Calculating the transfer functions, depending on the square root of the coherence and the psd's of the two channels
8. Dividing the arrays (z,h1,h2) in 4 parts to get length(fft) equal to length(transf)=32768
9. Giving the array's of (transf,f) a negative part, thereby doubling their length=32768. This piece was written to test a way to give the absolute values of the transferfunctions a negative side, so that they can be multiplied with the fft's to correct the signal for tilt of the seismometer. *But in this M-file the script is not yet written correctly, this is adjusted in the next M-file.*
10. Calculating the corrected values (rzc & rh2c) in 4 parts
11. Converting back from fft to time series
12. Calculating csd, psd, complex (root of) the coherence function, transfer functions for corrected z & h2 (zc&h2c)
13. Dividing the arrays (zc,h2c) in 4 parts to get length(fft) equal to length(transf)=32768. *Used "itry" again which messed up the script, this is corrected in the next M-file.*
14. Giving the array's of (transf,f) a negative part, thereby doubling their length=32768
15. Calculating the corrected values (rzcc) in 4 parts
16. Converting back from fft to time series & addition of 4 parts to form one chunk again
17. Plotting
18. Layout plotting

- **t19_totalcorrectOSN.m**

This M-file attempts to calculate and plot the Time Series and PSD of Z, H2 and:

- Z'= Z corrected for coherence with H1 using a transferfunction;
- H2'= H2 corrected for coherence with H1 using a transferfunction;
- Z"= Z' corrected for coherence with H2' using a transferfunction.

It is similar as the previous file only some corrections have been made in the script. The corrections are indicated with *this font*.
This happens in the following steps:

1. Open files with data of channels z,h1,h2
2. Read data in chunks of 1.8 hours
3. *Plotting time series of original z*

4. Set parameters
5. Calculating csd
6. Calculating psd (h2,h1,z) and *smoothed psd (z)*
7. Calculating the complex (root of) the coherence function
8. Calculating the transfer functions
9. Dividing the arrays (z,h1,h2) in 4 parts to get length(fft) equal to length(transf)=32768
10. *Giving the array's of (transf,f) a negative part, thereby doubling their length=32768*
    This piece was written to test a way to the absolute values of the transferfunctions a negative side, so that they can be multiplied with the fft's to correct the signal for tilt of the seismometer.
11. Calculating the corrected values (rzc & rh2c) in 4 parts
12. Converting back from fft to time series
13. Addition of 4 parts to form one chunk again
14. Calculating csd, psd, complex (root of) the coherence function, transfer functions for corrected z & h2 (zc&h2c)
15. *In case we want to plot smoothed corrected psd's*
16. Dividing the arrays (zc,h2c) in 4 parts to get length(fft) equal to length(transf)=32768
17. Giving the array's of (transf,f) a negative part, thereby doubling their length=32768
18. Calculating the corrected values (rzcc) in 4 parts
19. Converting back from fft to time series & addition of 4 parts to form one chunk again
20. *Smoothed corrected psd (zcc)*
21. *Multiply by -1, because plotting shows overturned curves*
22. Plotting
23. Layout plotting

- **t20_testnorm.m**

Little program written by Ralph, to test the normalisation of the psd function, by breaking it up into smaller steps. This can be used instead of the PSD function in the correction M-files 20_totalcorrect.m to avoid switching back and forth between the Time Domain and The Frequency domain.
This way we stay in the Frequency Domain.

- **t21_OSNwindcorZ.m**

This M-file attempts to calculate and plot PSD of Z and Z'. These are the PSD of the original Z and the PSD of Z corrected for coherent noise with H2. This is shorter than the corrections done in t19, and therefore hopefully this will work since there is less chance of making mistakes in the script. To calculate the PSD and apply the transferfunctions to it, we don't use the MATLAB psd function. Instead we take this function to pieces (described also in 20_testnorm.m) and apply the transferfunctions to overlapping windows. But something is still going wrong because the psd of corrected Z' still has still a higher noise level than the original Z...
A possible reason might be the difference in windows used; different from the paper used for these calculations. The following steps are taken:
1.open files with data of channels z, h1, h2
2.read data in chunks of 1.8 hours
3.set parameters
4.calculating csd
5.calculating psd (h2,h1,z) and smoothed psd (z)
6.calculating the complex (root of) the coherence function
7.calculating the transfer functions

8.giving the array's of (transf,f) a negative part, thereby doubling their length=32768
9.calculating the fft over 13 overlapping hanning windows with overlap (z&h2)
10.calculating the corrected values (rzc)
11.smoothed corrected psd (rzc)
12.plotting
13.layout plotting

- **TF54000.m**

function [mag,phase,fout,kend]=TF54000(nfft)
This function outputs the magnitude, phase and frequency vectors, which define the inverse, transfer function of the borehole system used on the OSNPE. nfft is the length of the fft used in computing the psd. Multiplying the psd (in amplitude units) by this transfer function converts counts on output to acceleration in nm/sec^2 on input. It assumes that the data are sampled at 20sps. The system transfer function is flat to velocity below one Hz. The output vectors are (nfft-1) long since the DC value is excluded. (Converting from velocity to acceleration at DC is meaningless.)
kend is the index of the last point to be within 3dB of flat velocity response.
This script computes the transfer function for a KS54000 with a Reftek digitizer and antialias filter.

- **smooth_spec.m**

function [fnew,Pnew]= smooth_spec(f,P,NPD); smooth spectra to approximately even spacing in log f USAGE: [fnew,Pnew]= smooth_spec(f,P,NPD);
INPUT:
    f column vector of frequencies (Hz)
    P column matrix of one or more power spectra
    NPD scalar specifying the number of frequencies per decade on output
OUTPUT
    fnew column vector containing the new frequencies
    Pnew column matrix containing the smoothed power spectra
Divide the frequency band into distinct regions that are evenly spaced in log10 f, with NPD frequencies per decade. Within each of these regions take the median of all power estimates to get Pnew and the median of all the frequencies to get fnew.

- **getspecdata_template1.m**
- **nhnm_avd**
- **nlnm_avd**

Results

- **t02s_tomgetspecdata_template1.m**
- **t03s_irenetest.m**
- **t04_puremfile.m**
- **t05_commentfile.m**

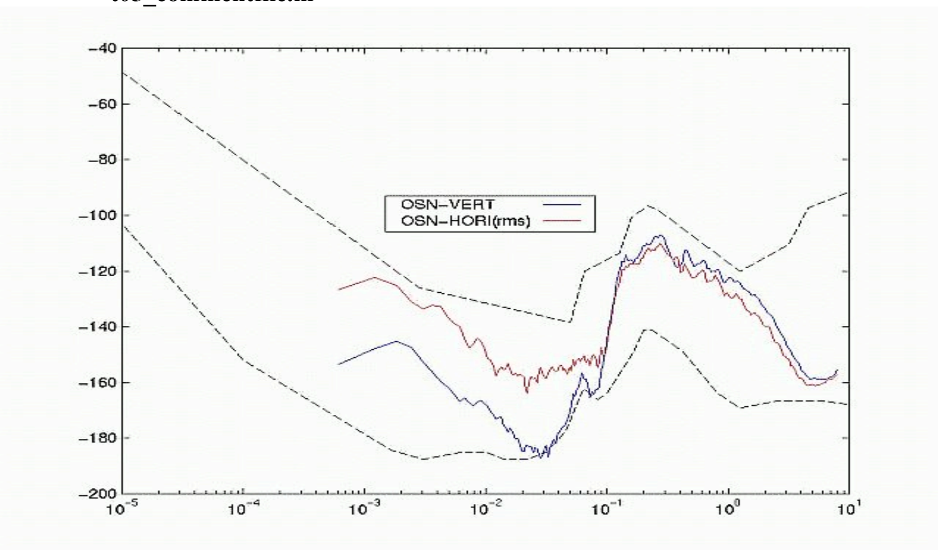


Fig.T10: PSD plot of vertical (Z) and horizontal (rms H) spectra. OSN data, day 62, borehole seismometer. Chunk 1 (1 chunk is approximately 1.8 hours of data).

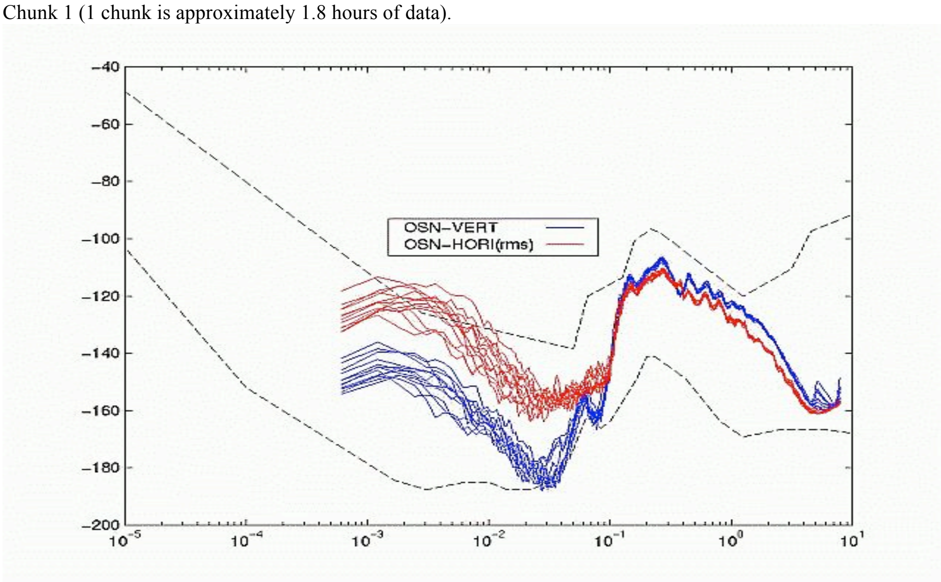


Fig.T11: PSD plot of vertical (Z) and horizontal (rms H) spectra. OSN data, day 62, borehole seismometer. All (13) chunks (1 chunk is approximately 1.8 hours of data).

- **t06_timeplot.m**

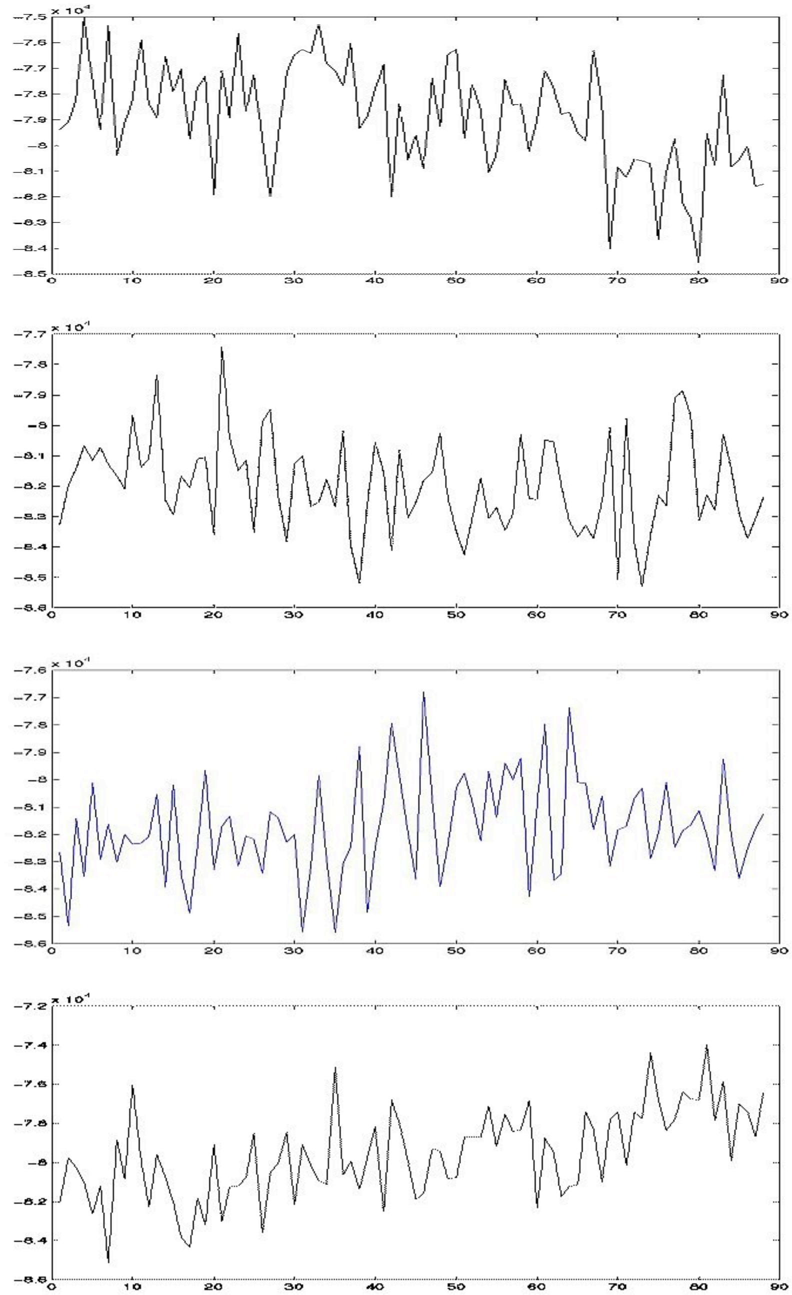


Fig. T12: Time series of vertical component with every 1500 point plotted. OSN data, day 62, borehole seismometer. Chunks 1, 2 , 3 and 4 of the vertical component.

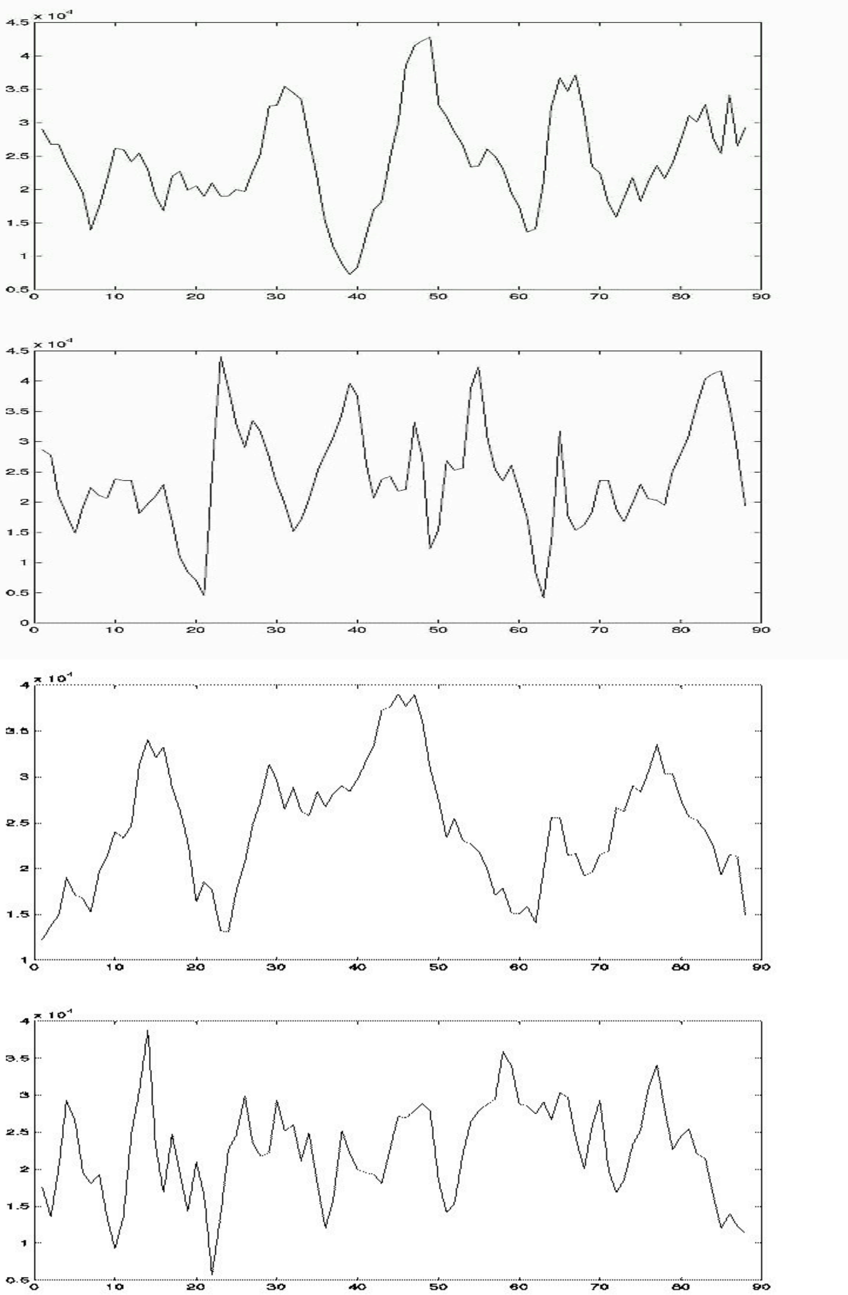


Fig. T13: Time series of rms horizontal components with every 1500 point plotted. OSN data, day 62, borehole seismometer. Chunks 1, 2 , 3 and 4 of the rms horizontal component.

- **t10_OSNcsd.m**

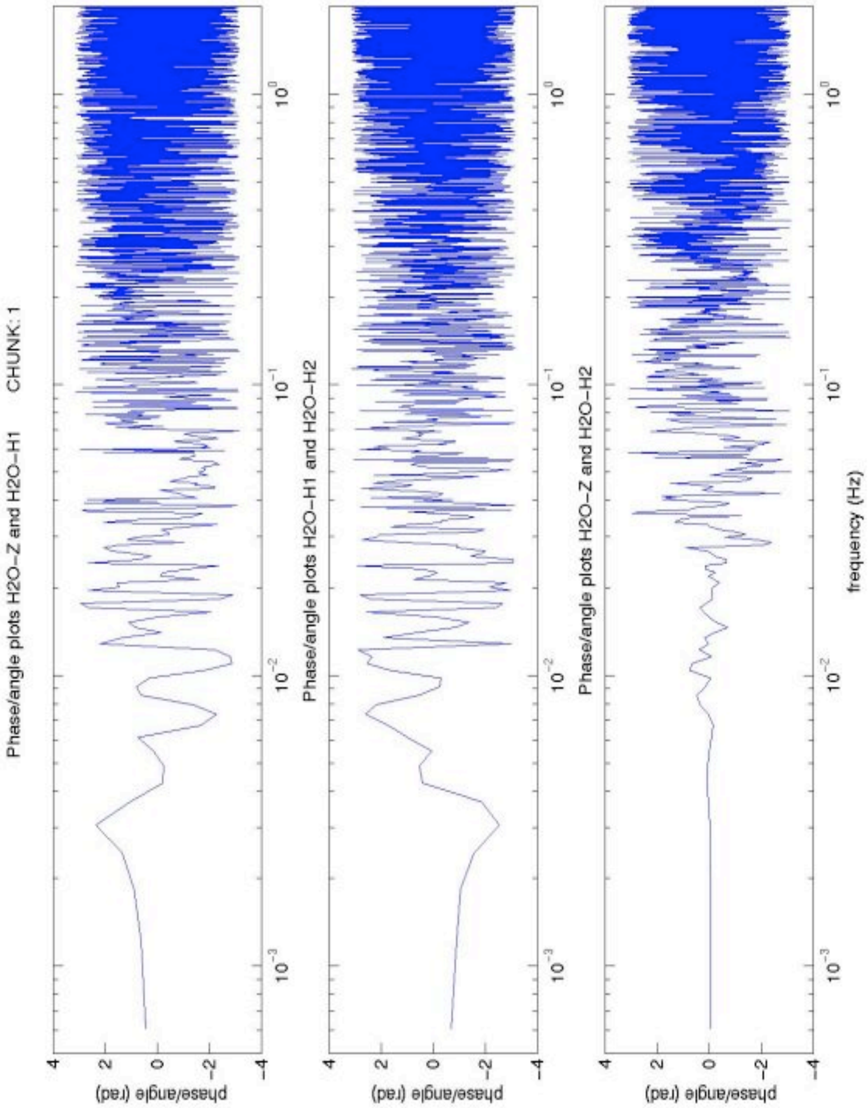


Fig. T14: CSD plots of Z+H1, H1+H2 and Z+H2. OSN data, day 62, borehole seismometer. Chunk 1.

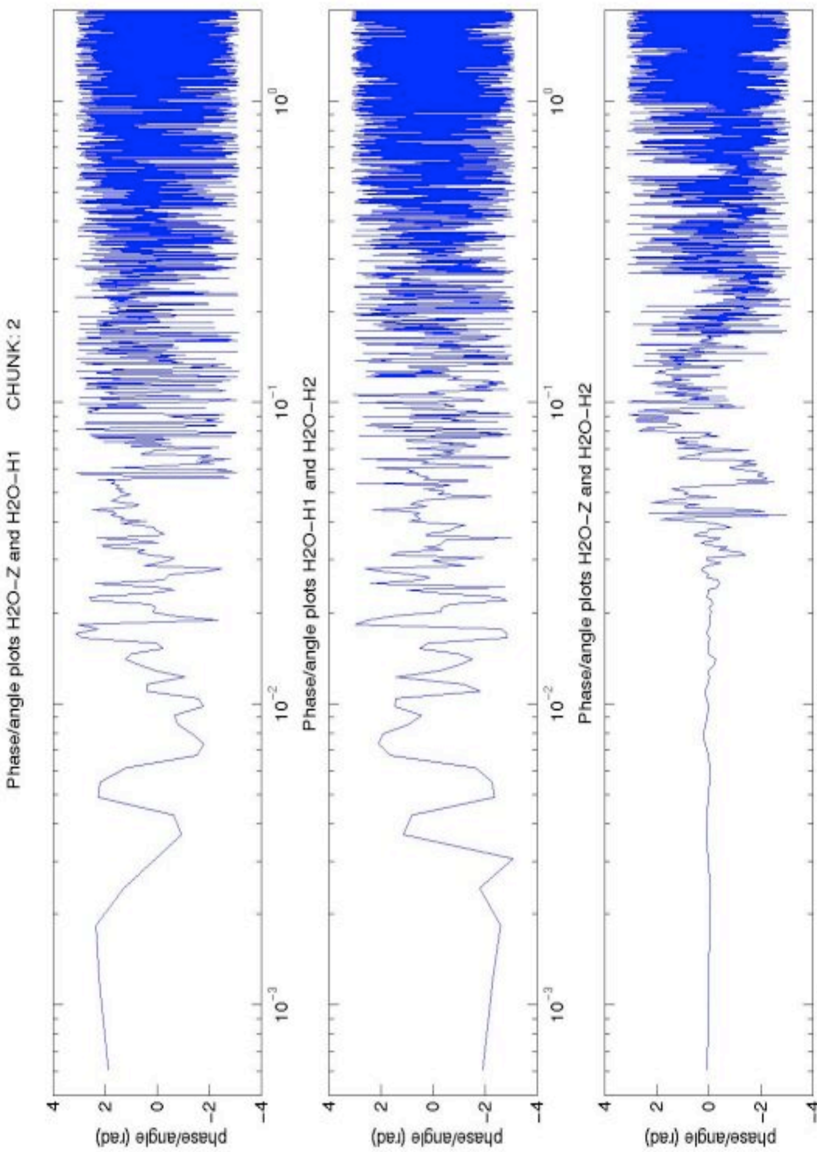


Fig. T15: CSD plots of Z+H1, H1+H2 and Z+H2. OSN data, day 62, borehole seismometer. Chunk 2.

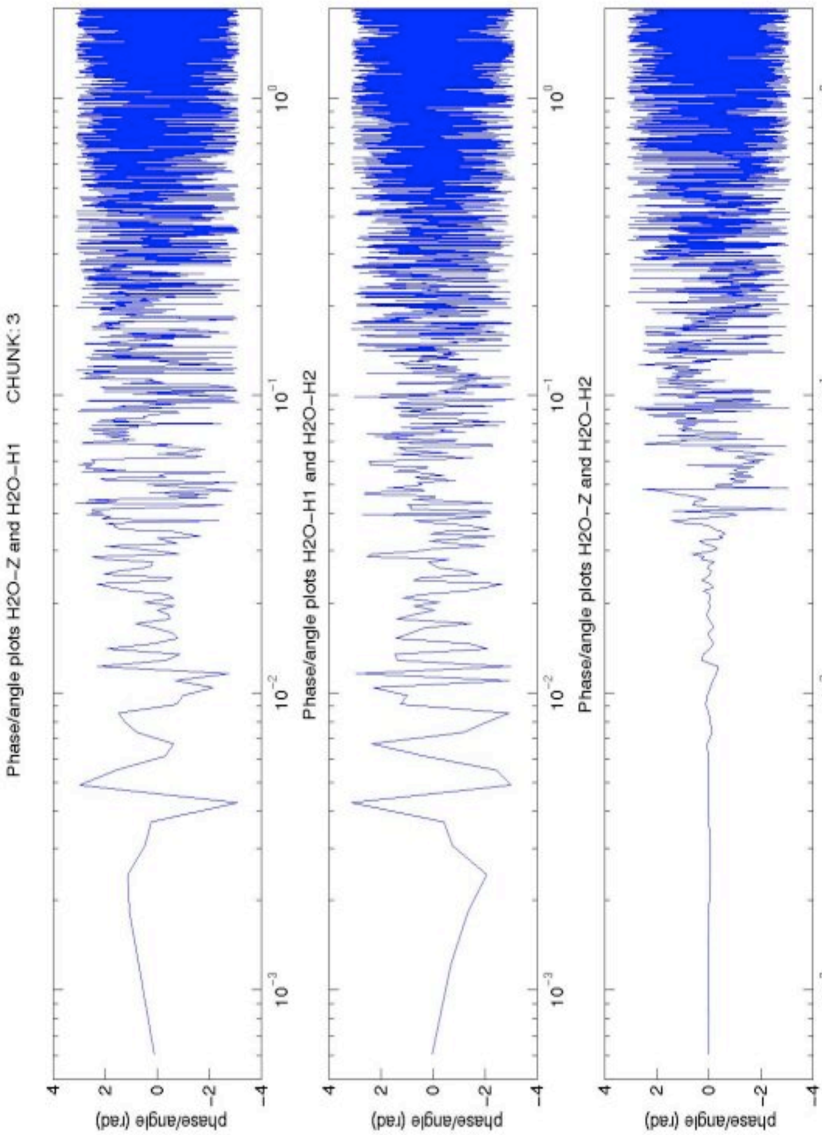


Fig. T16: CSD plots of Z+H1, H1+H2 and Z+H2. OSN data, day 62, borehole seismometer. Chunk 3.

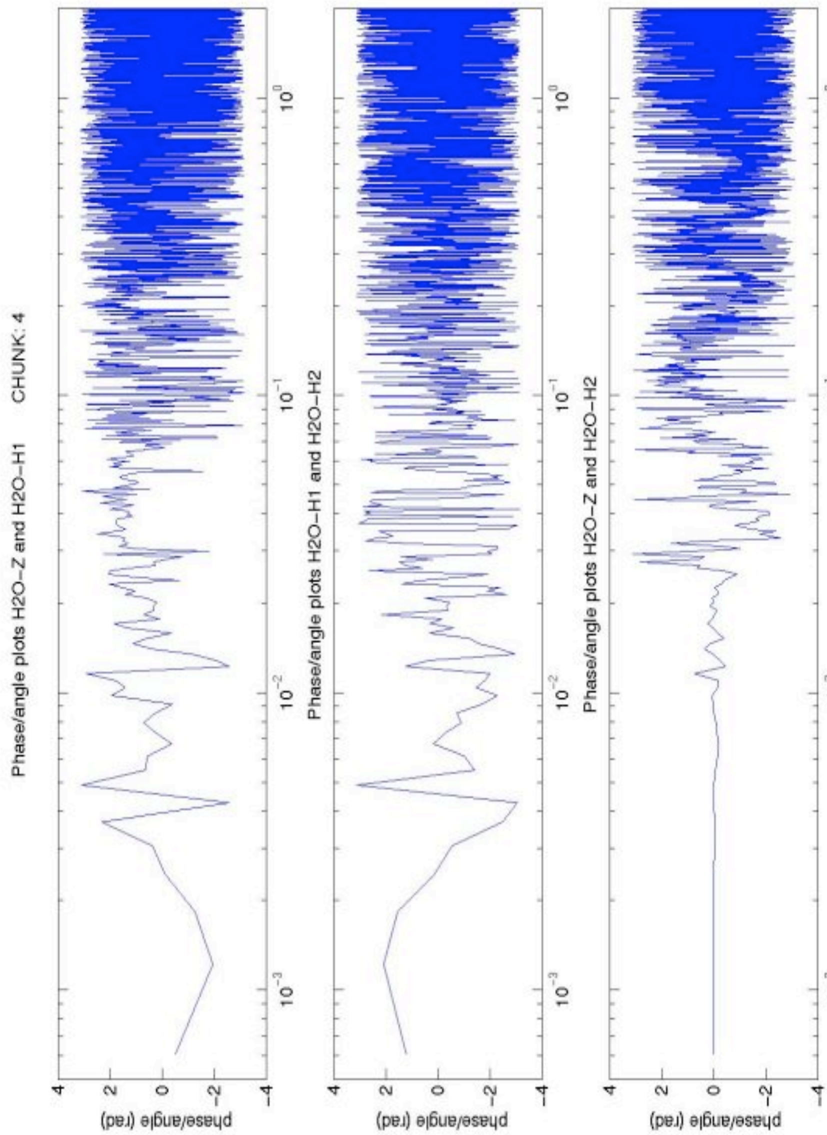


Fig. T17: CSD plots of Z+H1, H1+H2 and Z+H2. OSN data, day 62, borehole seismometer. Chunk 4.

- **t10_H2Ocsd.m**

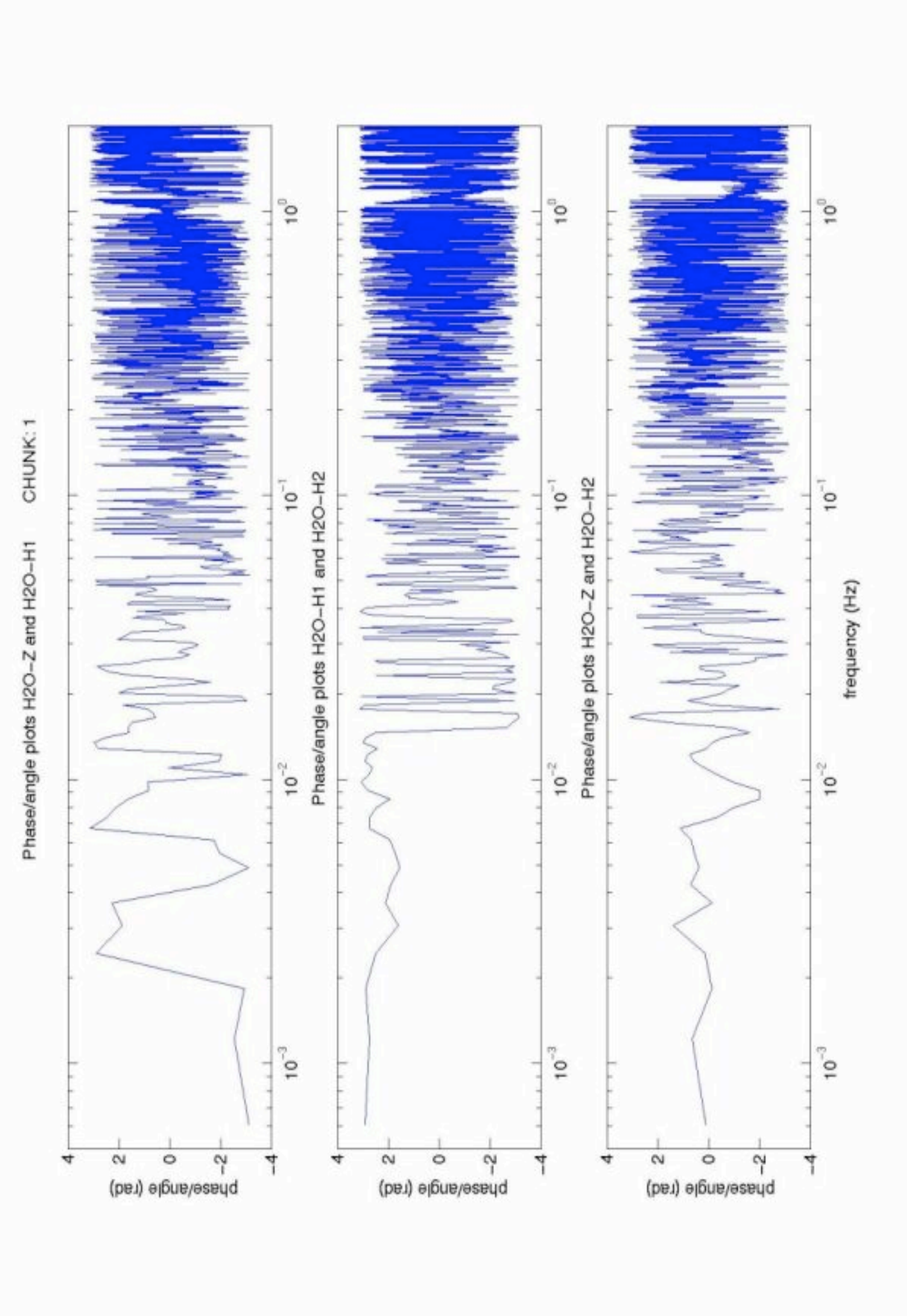


Fig. T18: CSD plots of Z+H1, H1+H2 and Z+H2. H2O data, day 121, buried seismometer. Chunk 1.

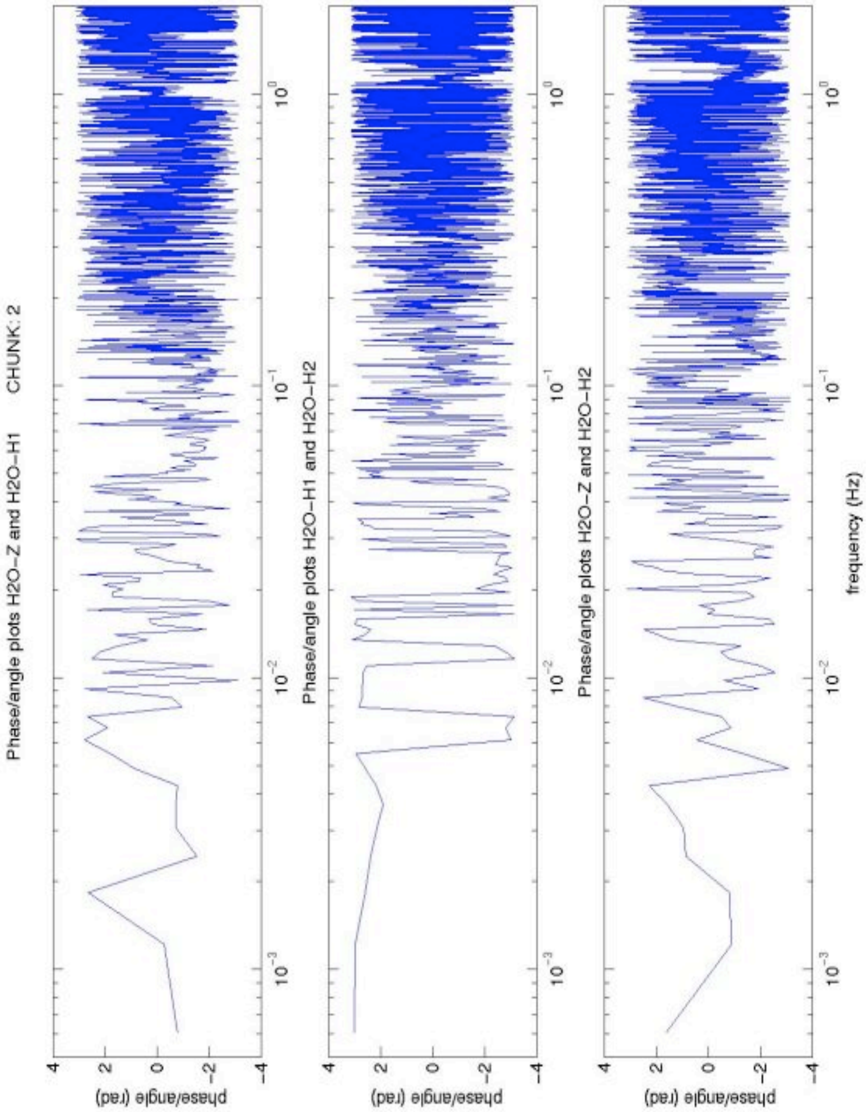


Fig. T19: CSD plots of Z+H1, H1+H2 and Z+H2. H2O data, day 121, buried seismometer. Chunk 2.

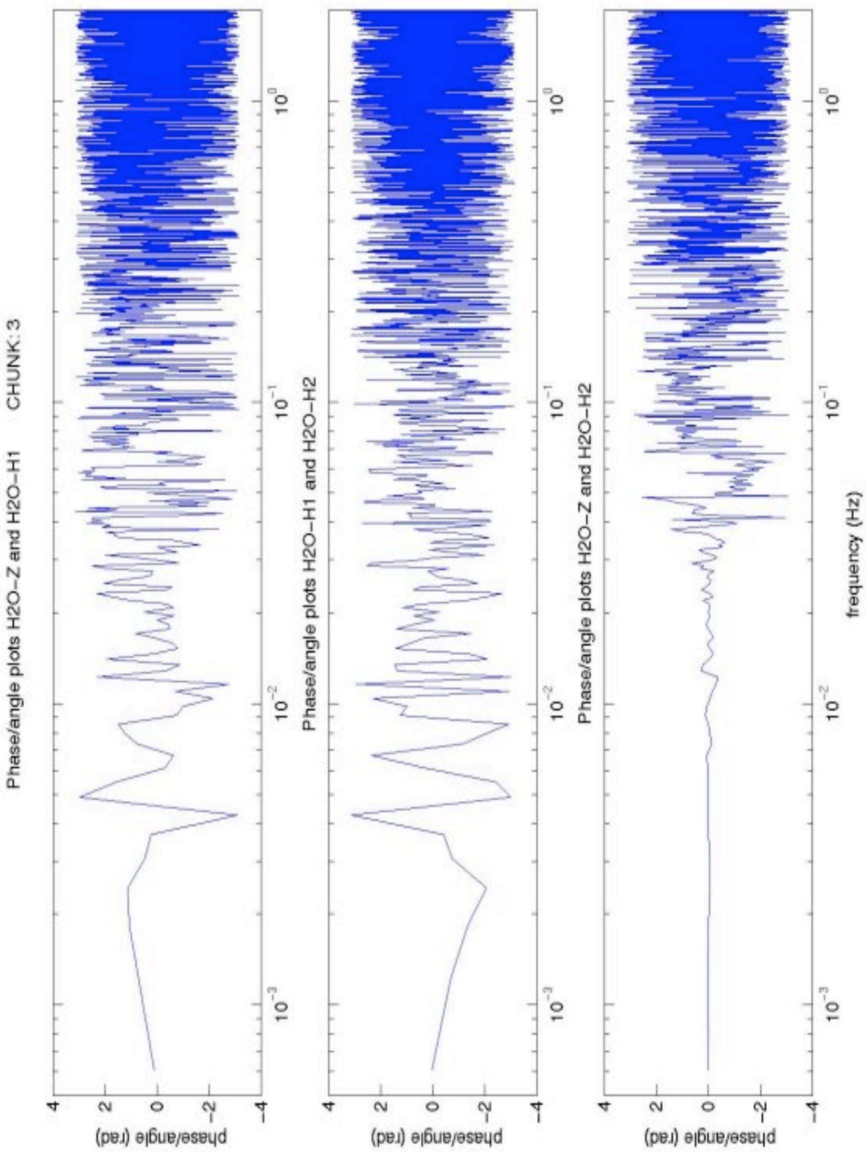


Fig. T20: CSD plots of Z+H1, H1+H2 and Z+H2. H2O data, day 121, buried seismometer. Chunk 3.

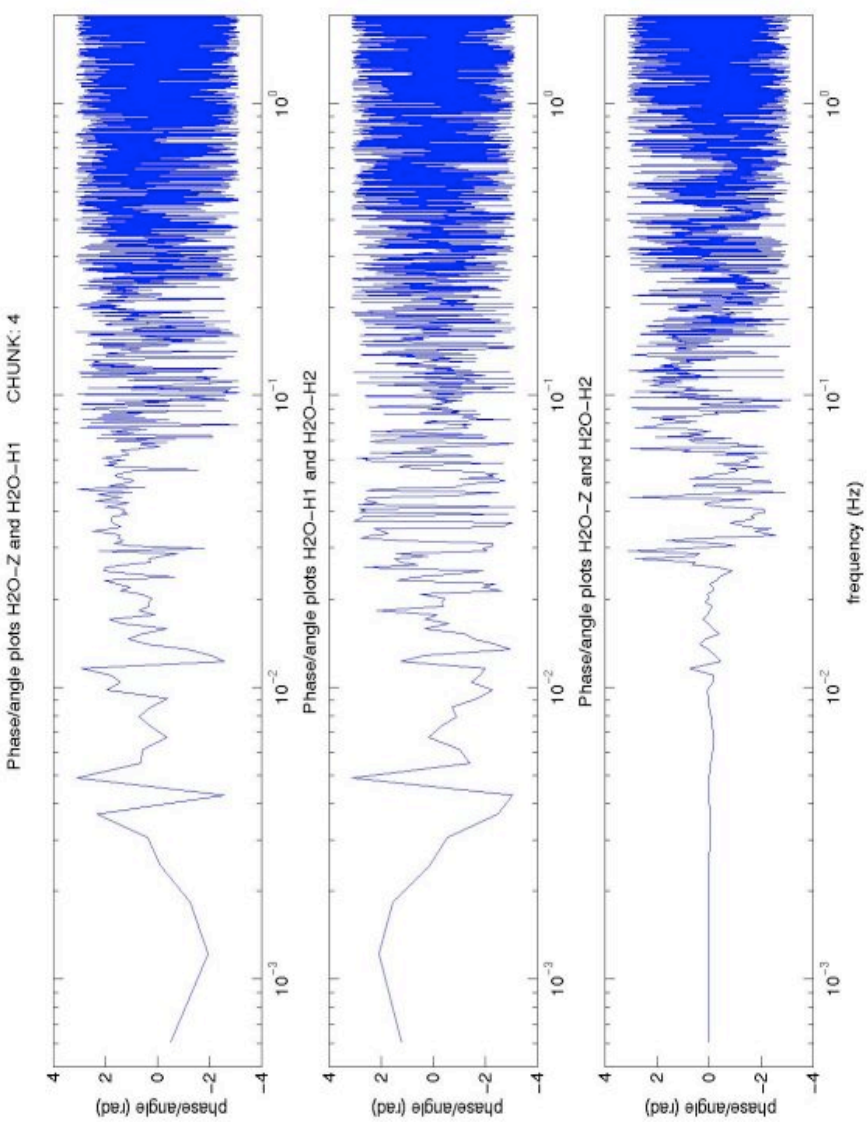


Fig. T21: CSD plots of Z+H1, H1+H2 and Z+H2. H2O data, day 121, buried seismometer. Chunk 4.

- **t11_OSNpsd+coh.m**

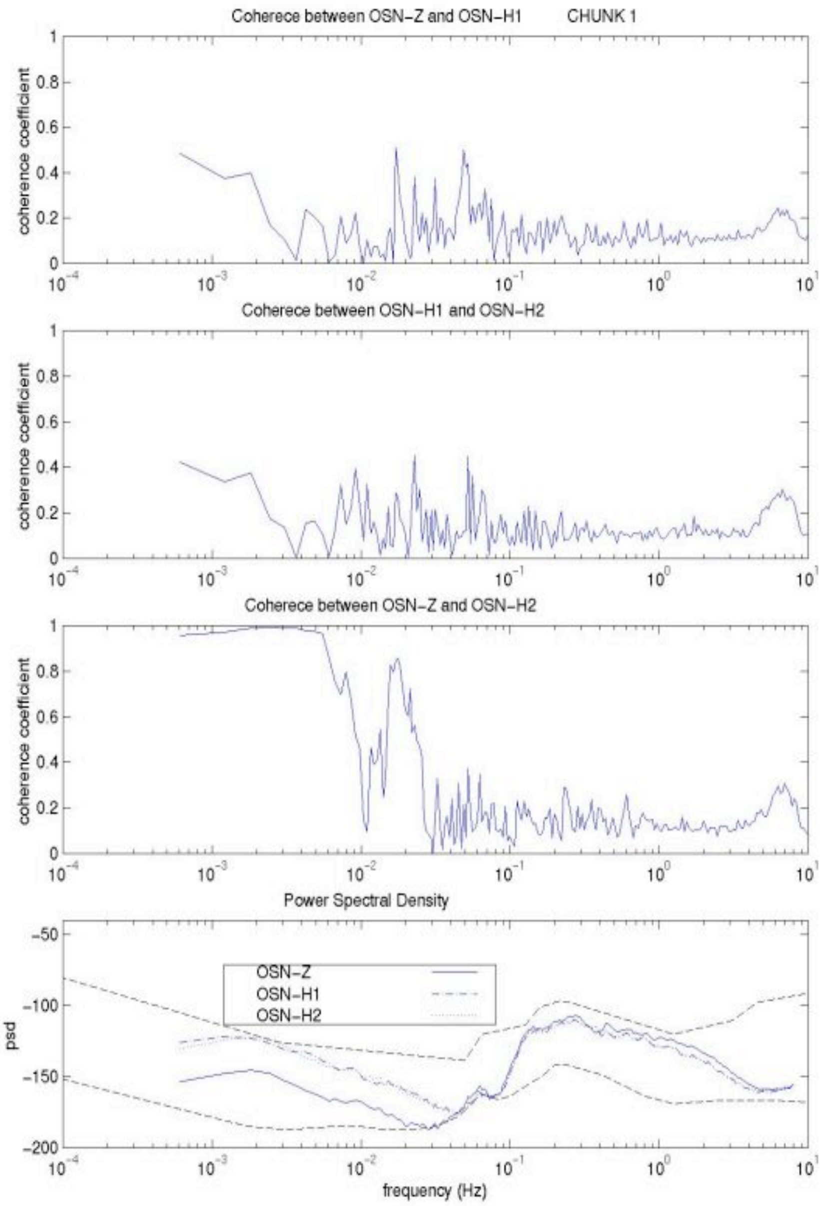


Fig. T22: Plots of coherence between Z+H1, H1+ H2 and Z+H2 and a plot of psd of Z, H1 and H2. OSN data, day 62, borehole seismometer. Chunk 1.

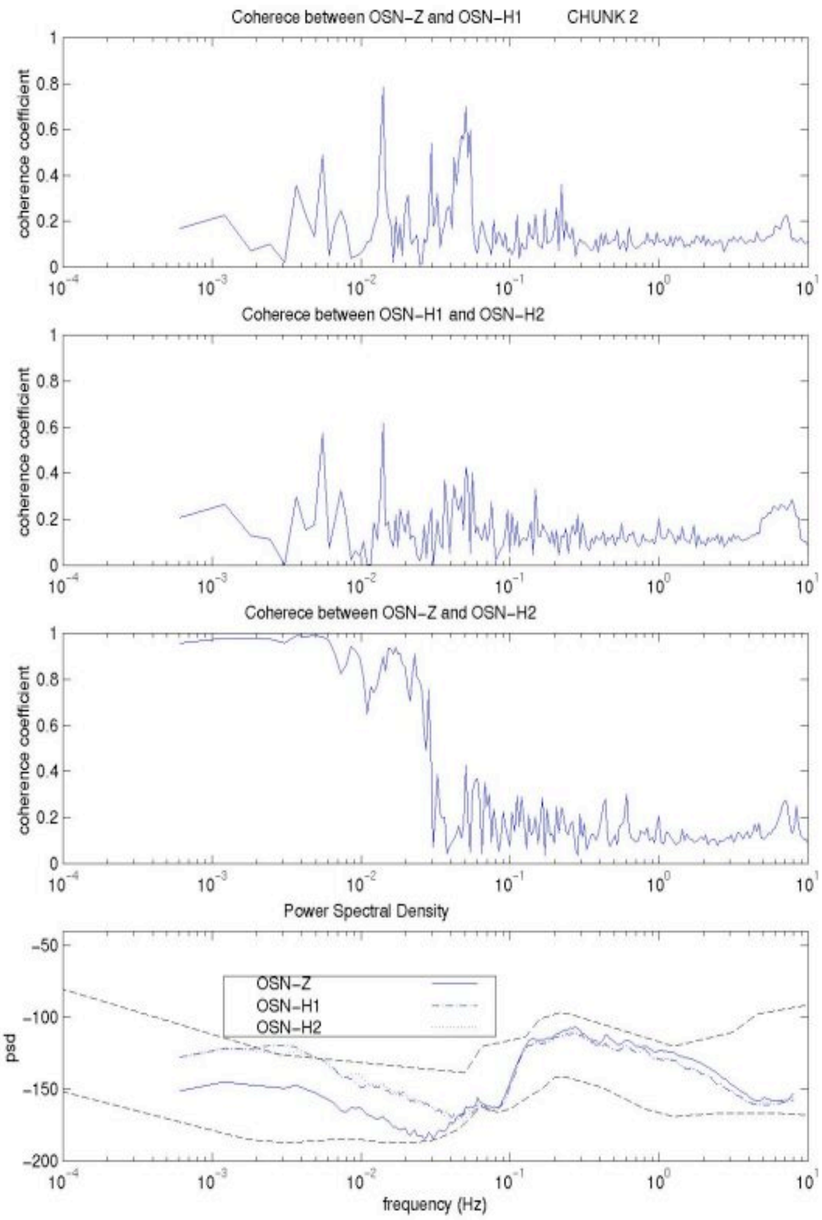


Fig. T23: Plots of coherence between Z+H1, H1+ H2 and Z+H2 and a plot of psd of Z, H1 and H2. OSN data, day 62, borehole seismometer. Chunk 2.

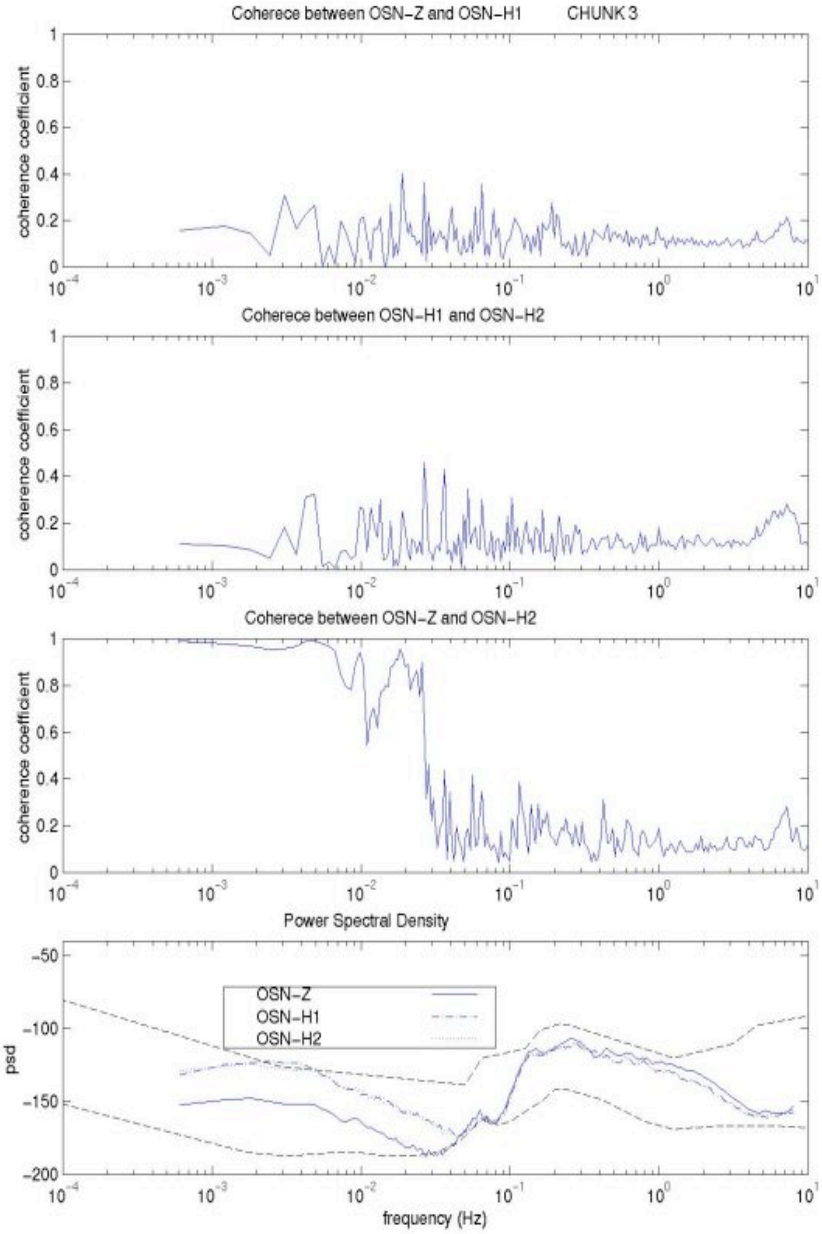


Fig. T24: Plots of coherence between Z+H1, H1+ H2 and Z+H2 and a plot of psd of Z, H1 and H2. OSN data, day 62, borehole seismometer. Chunk 3.

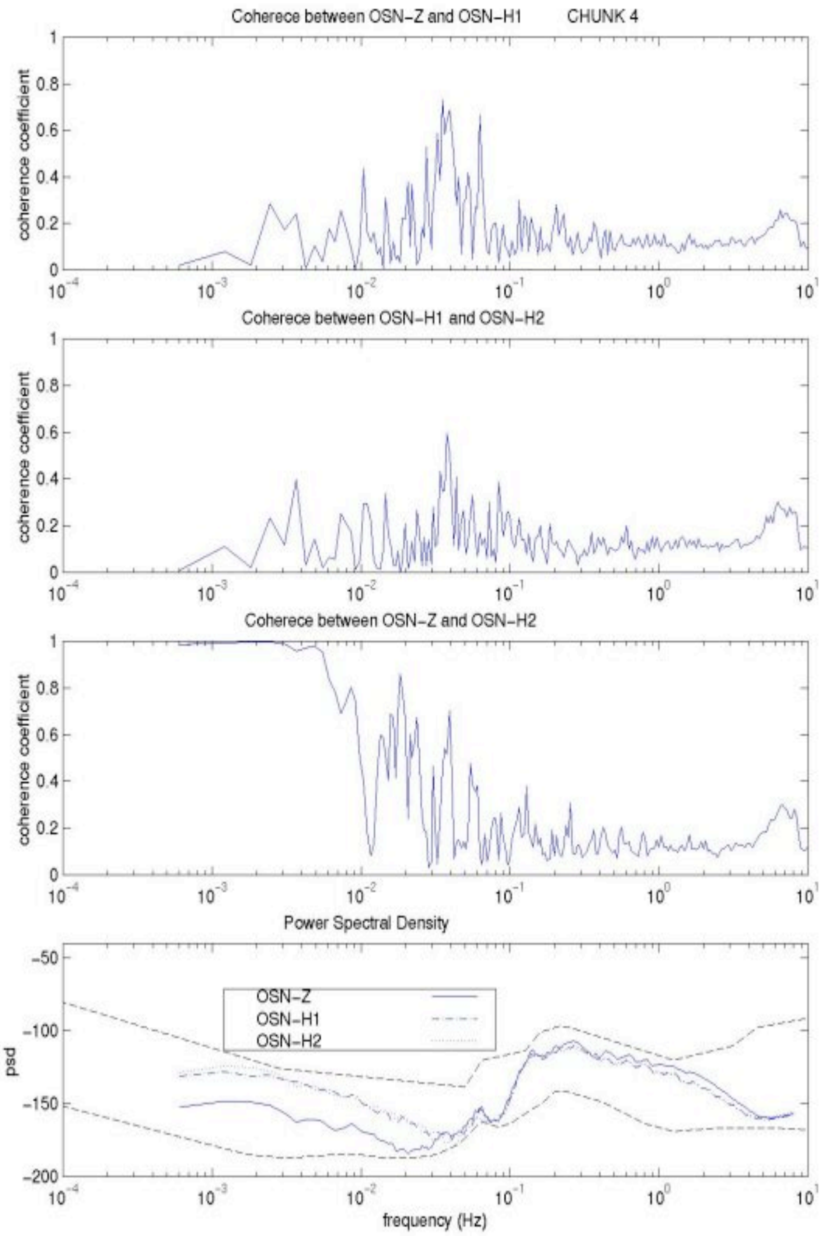


Fig. T25: Plots of coherence between Z+H1, H1+ H2 and Z+H2 and a plot of psd of Z, H1 and H2. OSN data, day 62, borehole seismometer. Chunk 4.

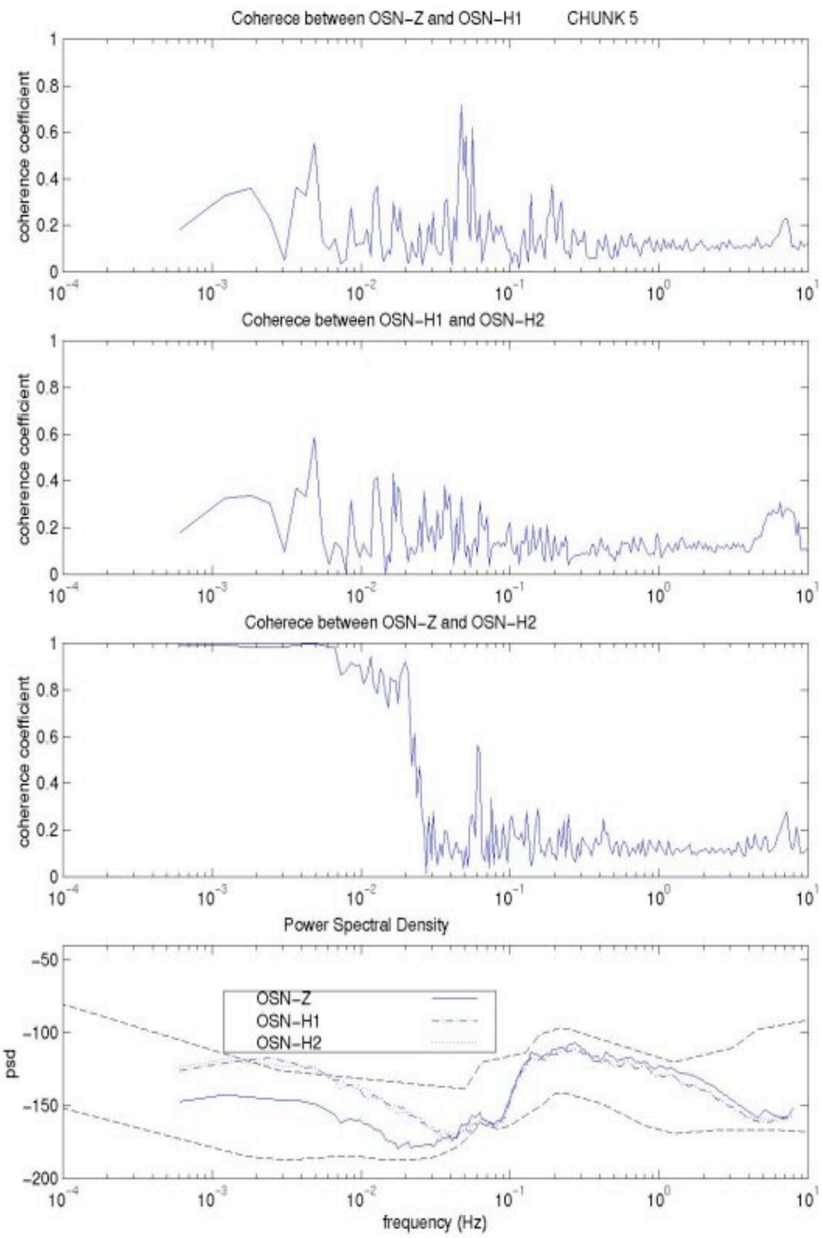


Fig. T26: Plots of coherence between Z+H1, H1+ H2 and Z+H2 and a plot of psd of Z, H1 and H2. OSN data, day 62, borehole seismometer. Chunk 5.

- **t12_H2Opsd+coh.m**

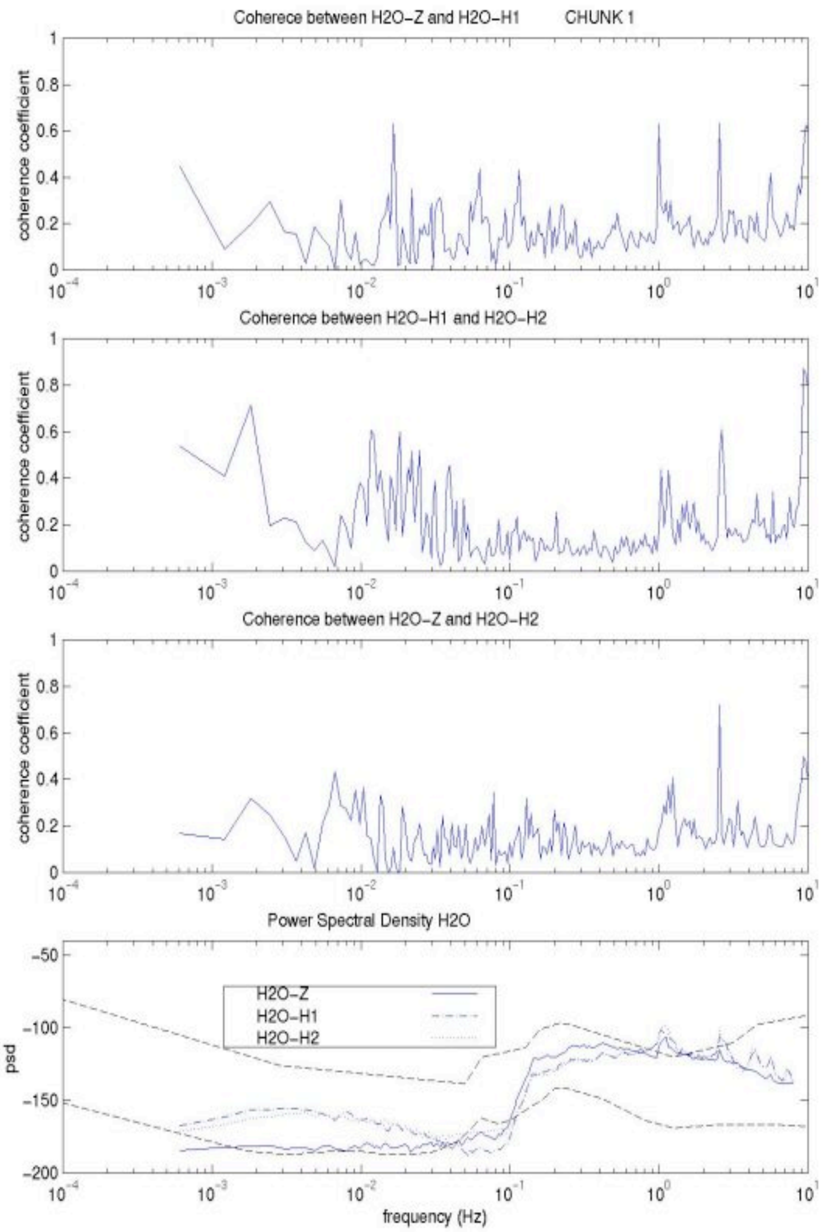


Fig. T27: Plots of coherence between Z+H1, H1+ H2 and Z+H2 and a plot of psd of Z, H1 and H2. H2O data, day 121, buried seismometer. Chunk 1.

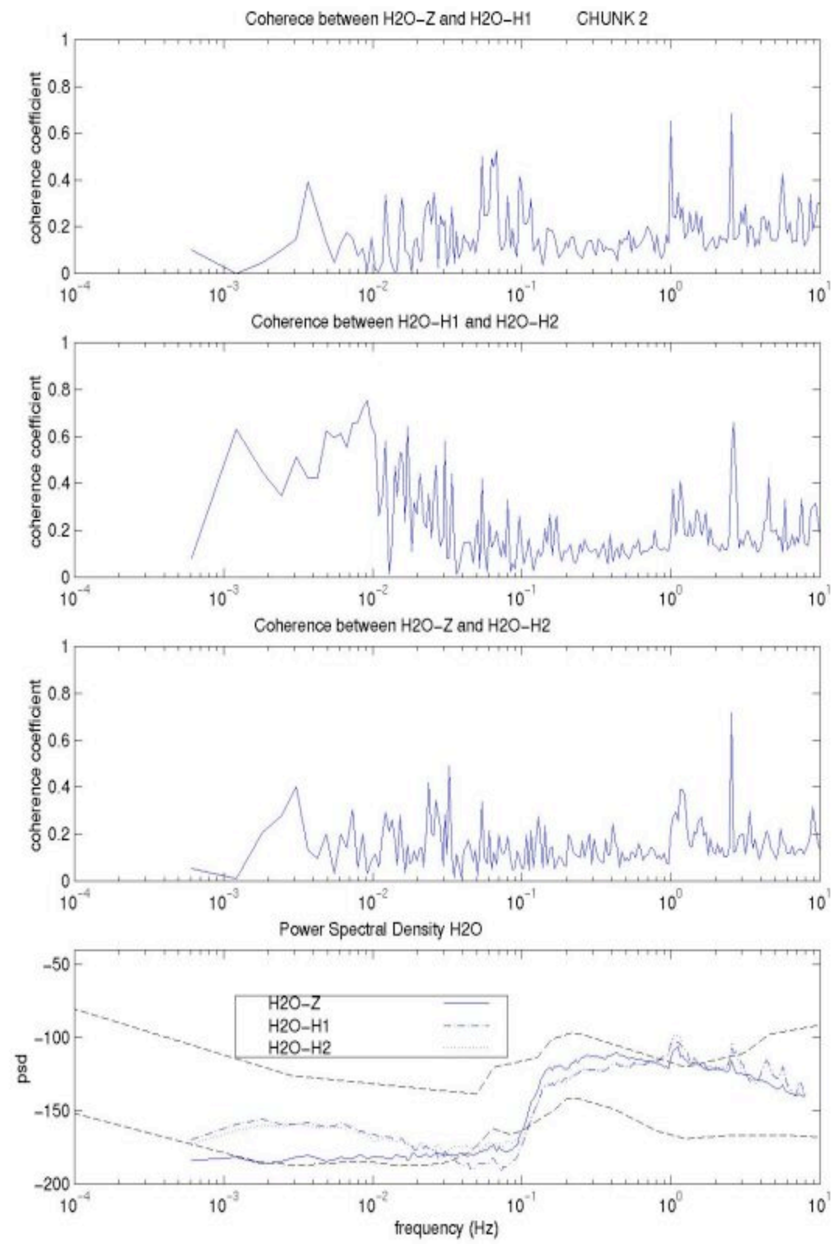


Fig. T28: Plots of coherence between Z+H1, H1+ H2 and Z+H2 and a plot of psd of Z, H1 and H2. H2O data, day 121, buried seismometer. Chunk 2.

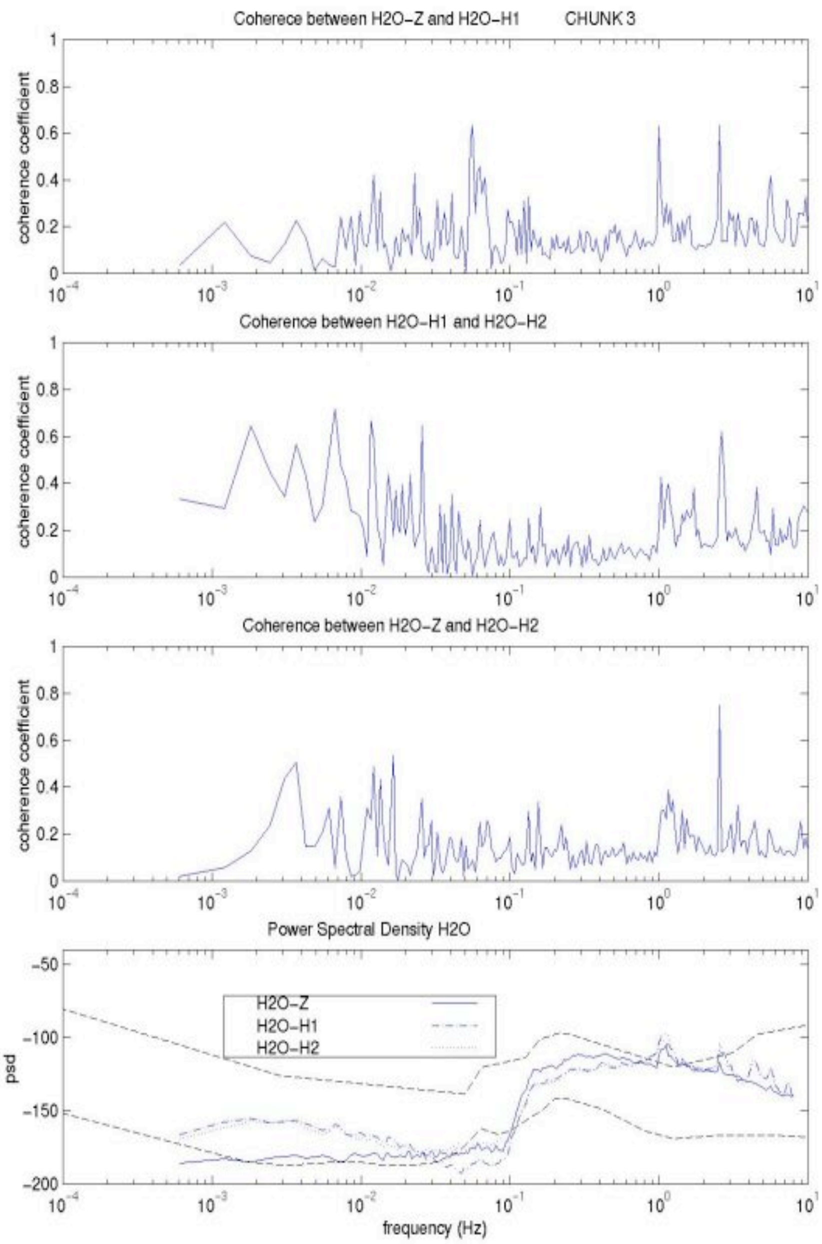


Fig. T29: Plots of coherence between Z+H1, H1+ H2 and Z+H2 and a plot of psd of Z, H1 and H2. H2O data, day 121, buried seismometer. Chunk 3.

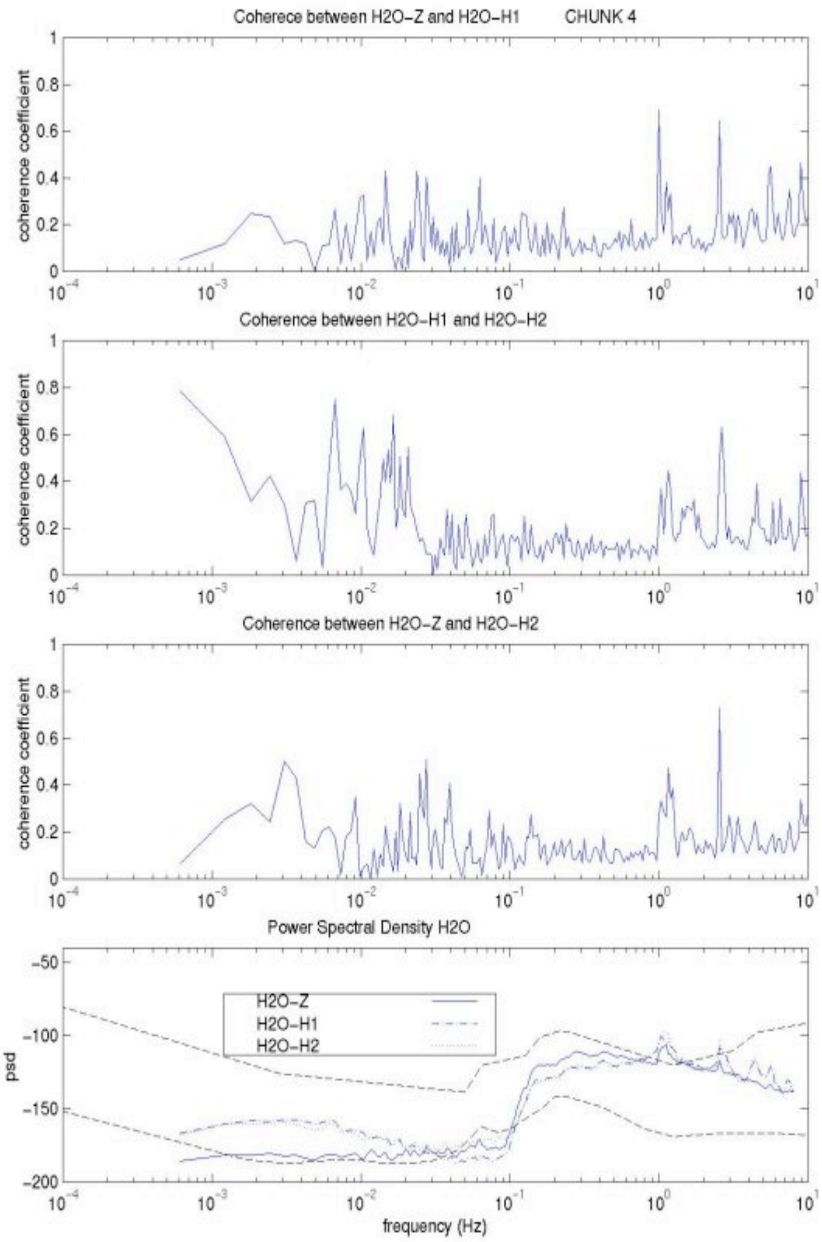


Fig. T30: **Plots of coherence between Z+H1, H1+ H2 and Z+H2 and a plot of psd of Z, H1 and H2. H2O data, day 121, buried seismometer. Chunk 4.**

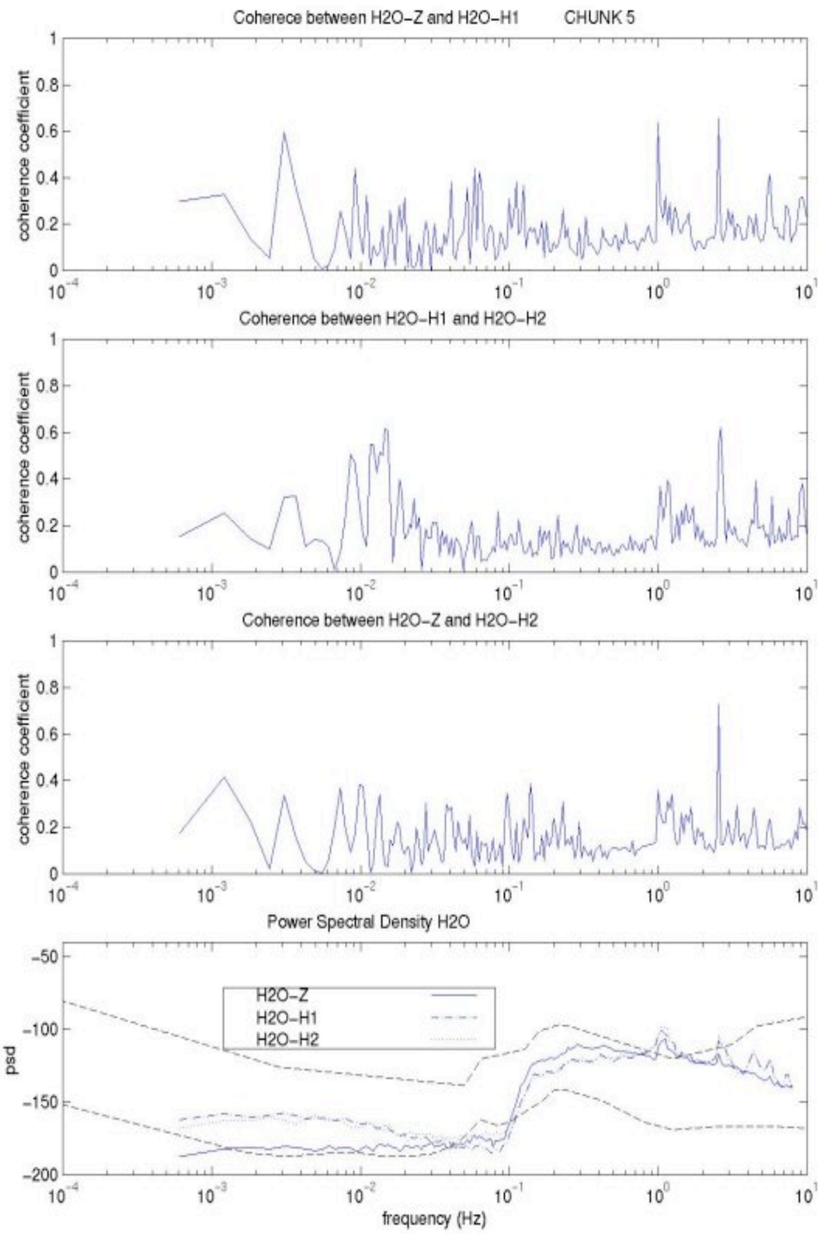


Fig. T31: **Plots of coherence between Z+H1, H1+ H2 and Z+H2 and a plot of psd of Z, H1 and H2. H2O data, day 121, buried seismometer. Chunk 5.**

- t13_OSNtransf.m

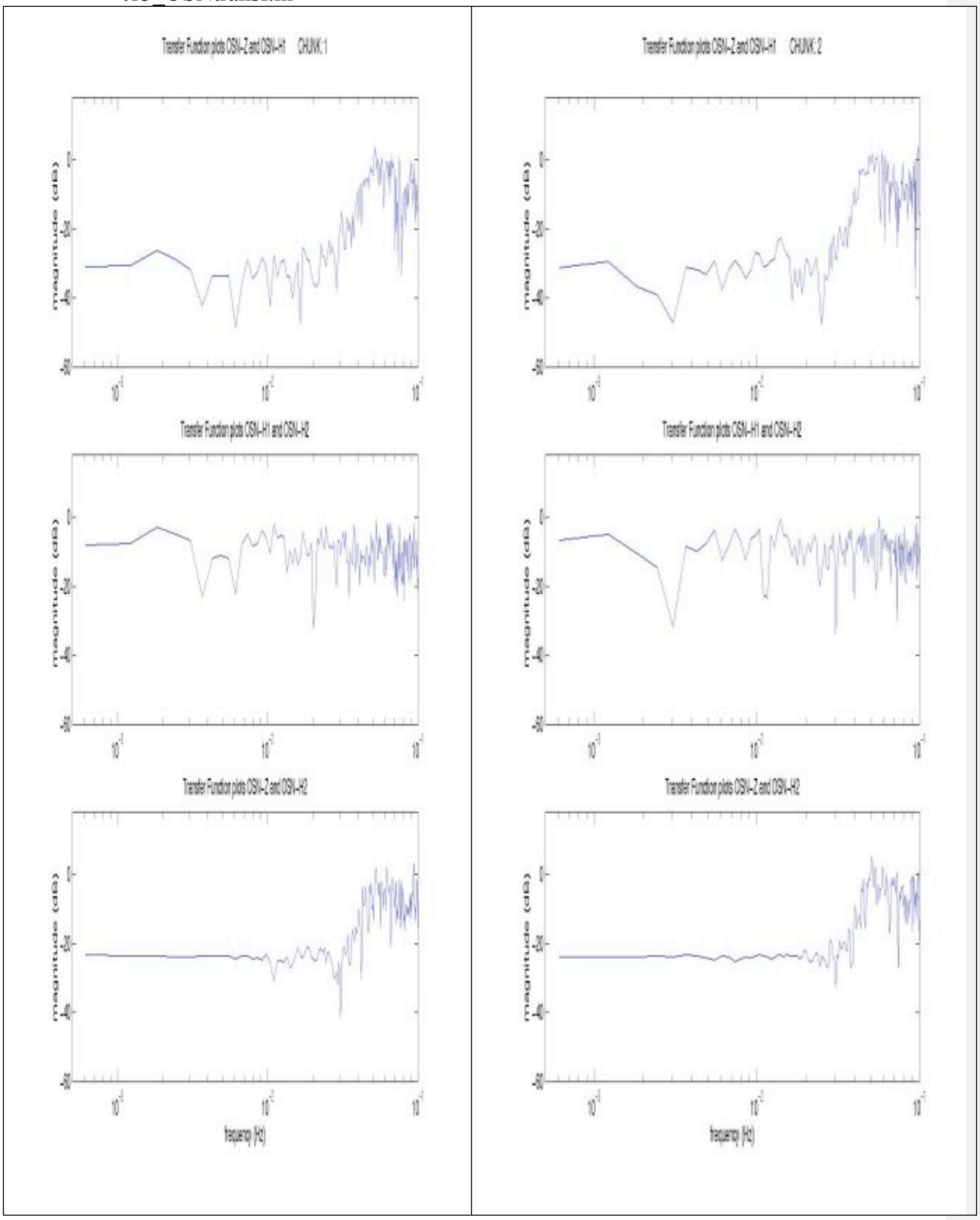


Fig. T32: **Plot of magnitude of transferfunctions (calculated with the MATLAB "cohere" option) between the seismic channels. OSN data, day 62, borehole seismometer. Chunks 1 and 2.**

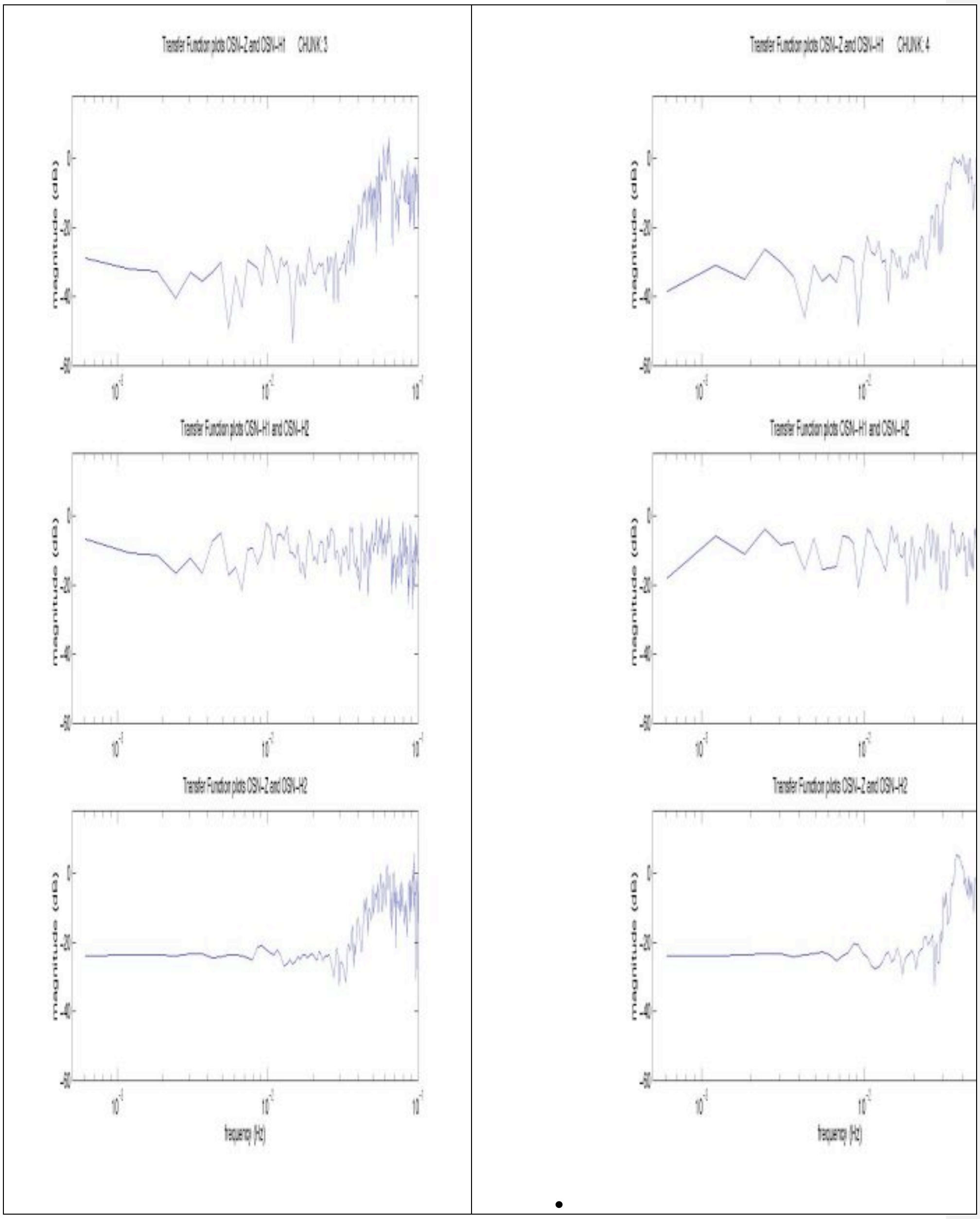


Fig. T33: **Plot of magnitude of transferfunctions (calculated with the MATLAB "cohere" option) between the seismic channels. OSN data, day 62, borehole seismometer. Chunks 3 and 4.**

- t14s_H2Otransf.m

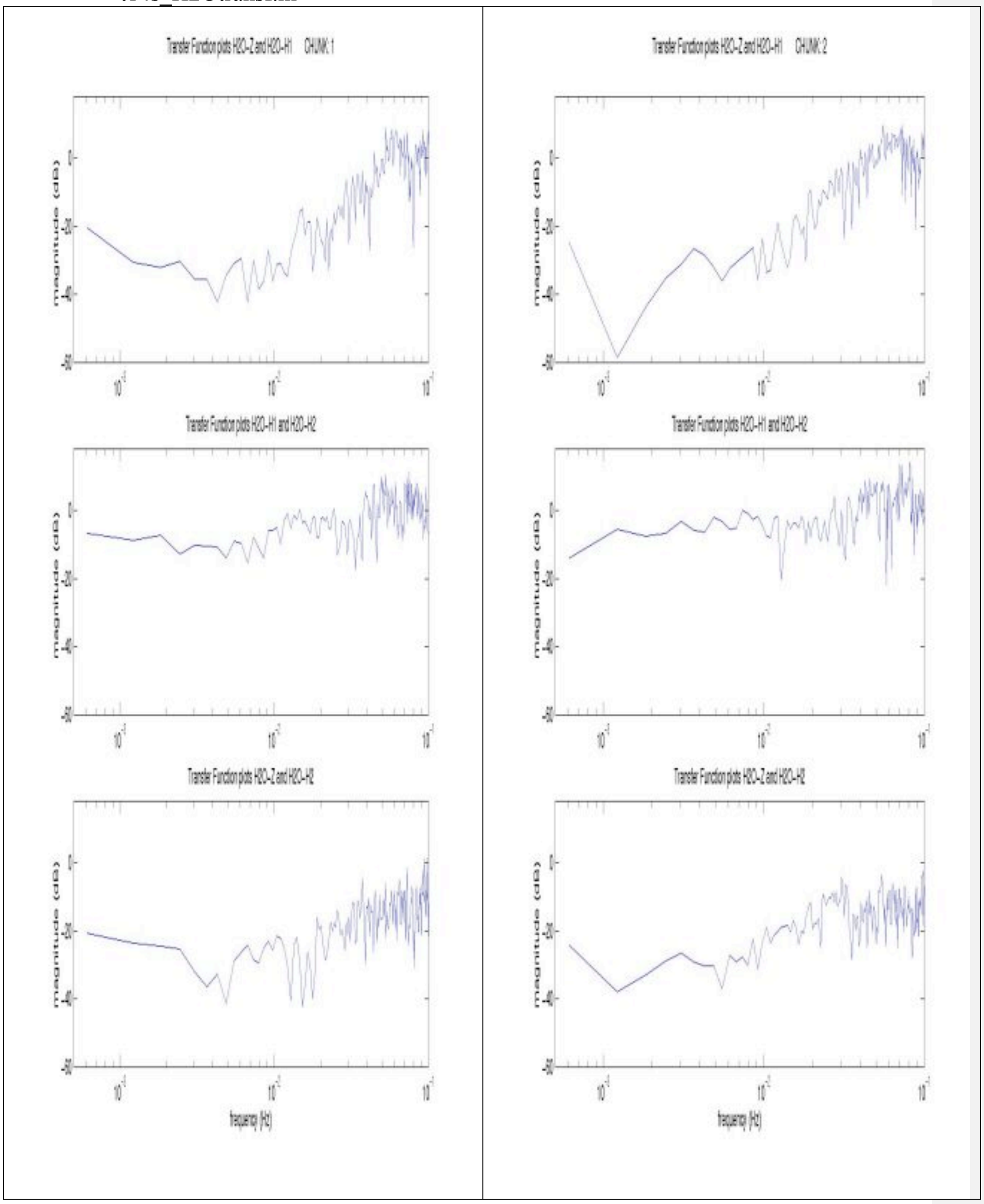


Fig. T34: **Plot of magnitude of transferfunctions (calculated with the MATLAB "cohere" option) between the seismic channels. H2O data, day 121, buried seismometer. Chunks 1 and 2.**

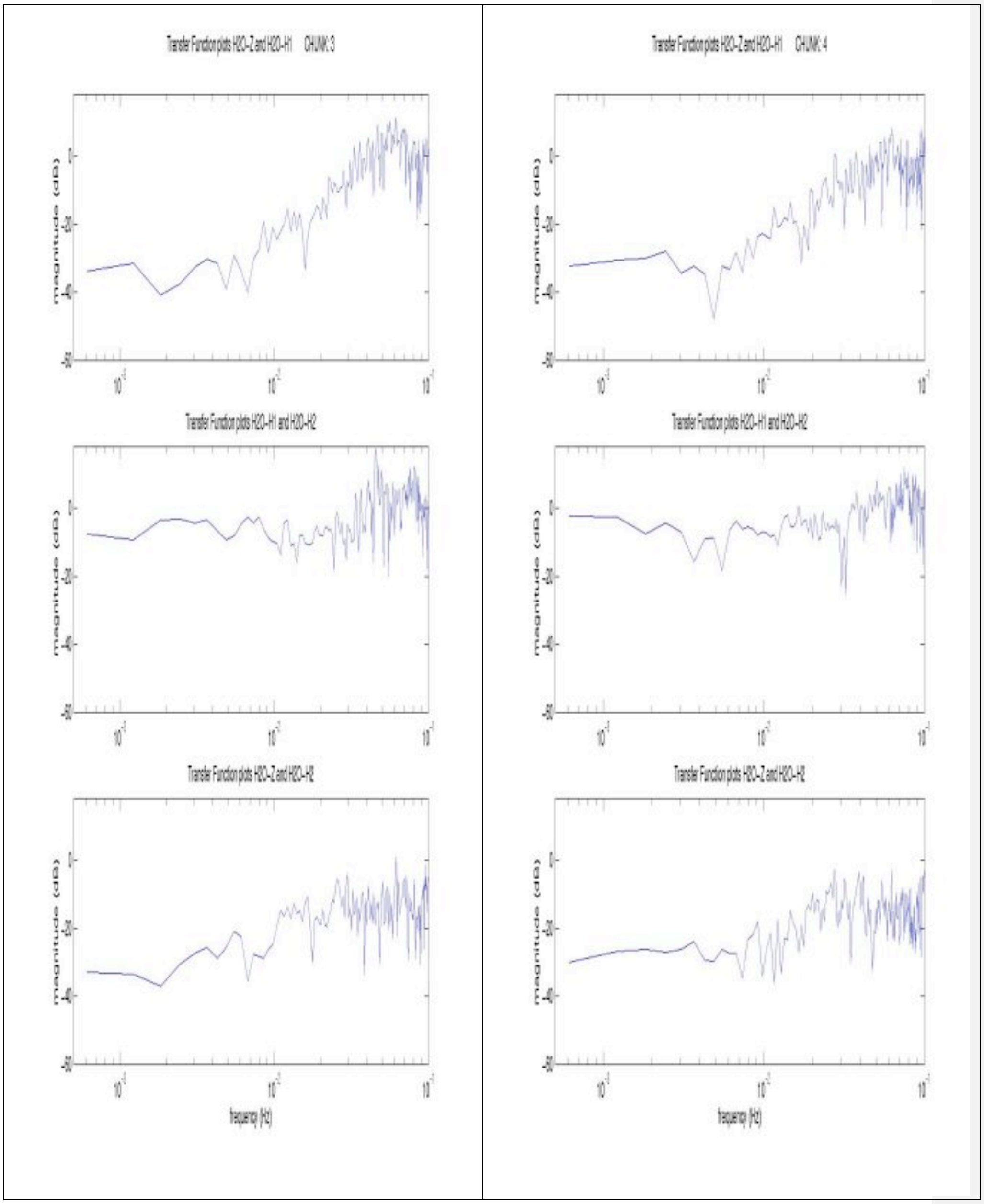


Fig. T35: **Plot of magnitude of transferfunctions (calculated with the MATLAB "cohere" option) between the seismic channels. H2O data, day 121, buried seismometer. Chunks 3 and 4.**

- **t15_easytransf.m**

- **t16_H2Otransfa.m**

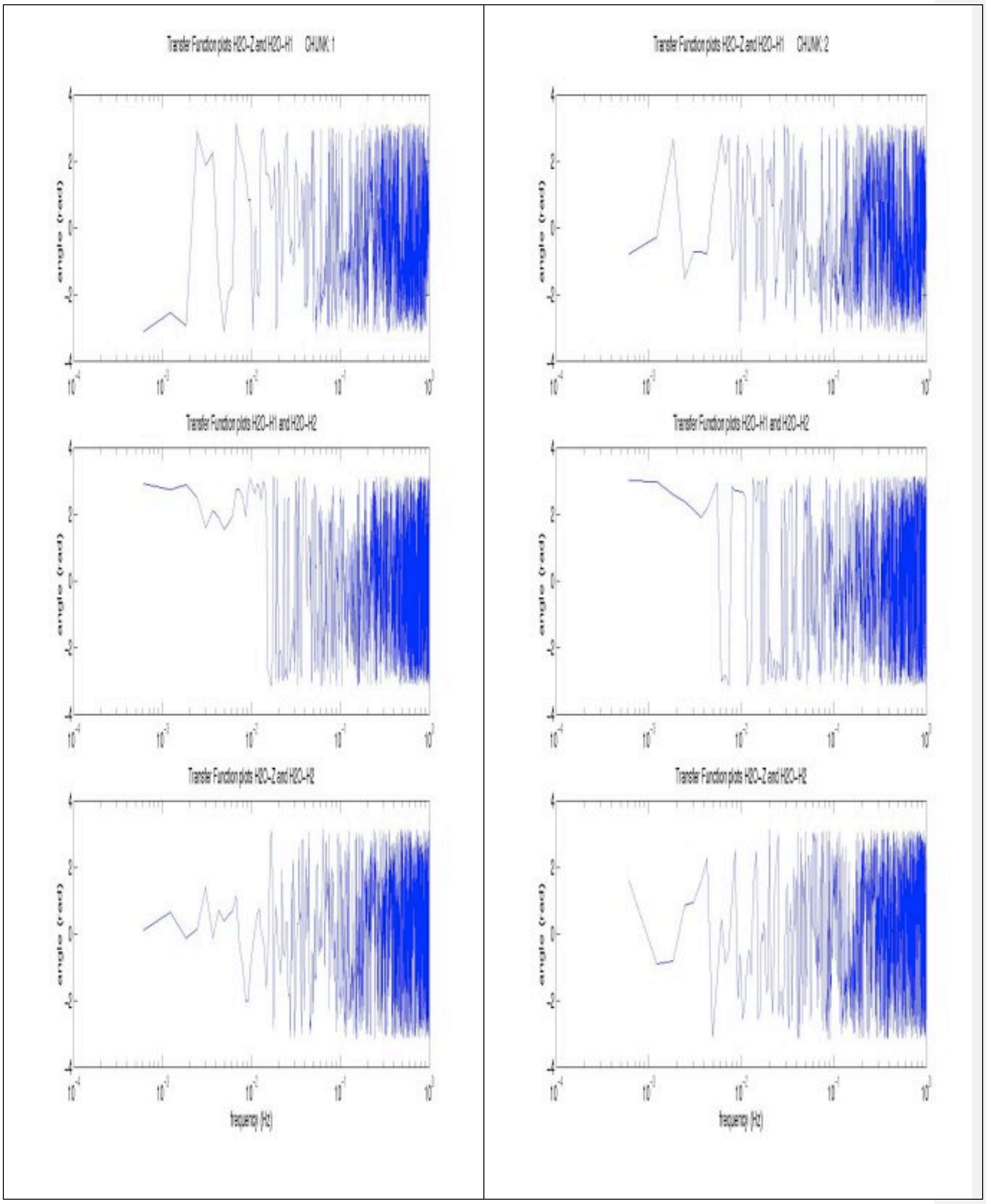


Fig. T36: Plot of the angle / phase of the transfer factions (calculated by using the theory of Crawford & Webb – using CSD and PSD to determine coherence) between the seismic channels. **H2O** data, day 121, buried seismometer. Chunks 1 and 2.

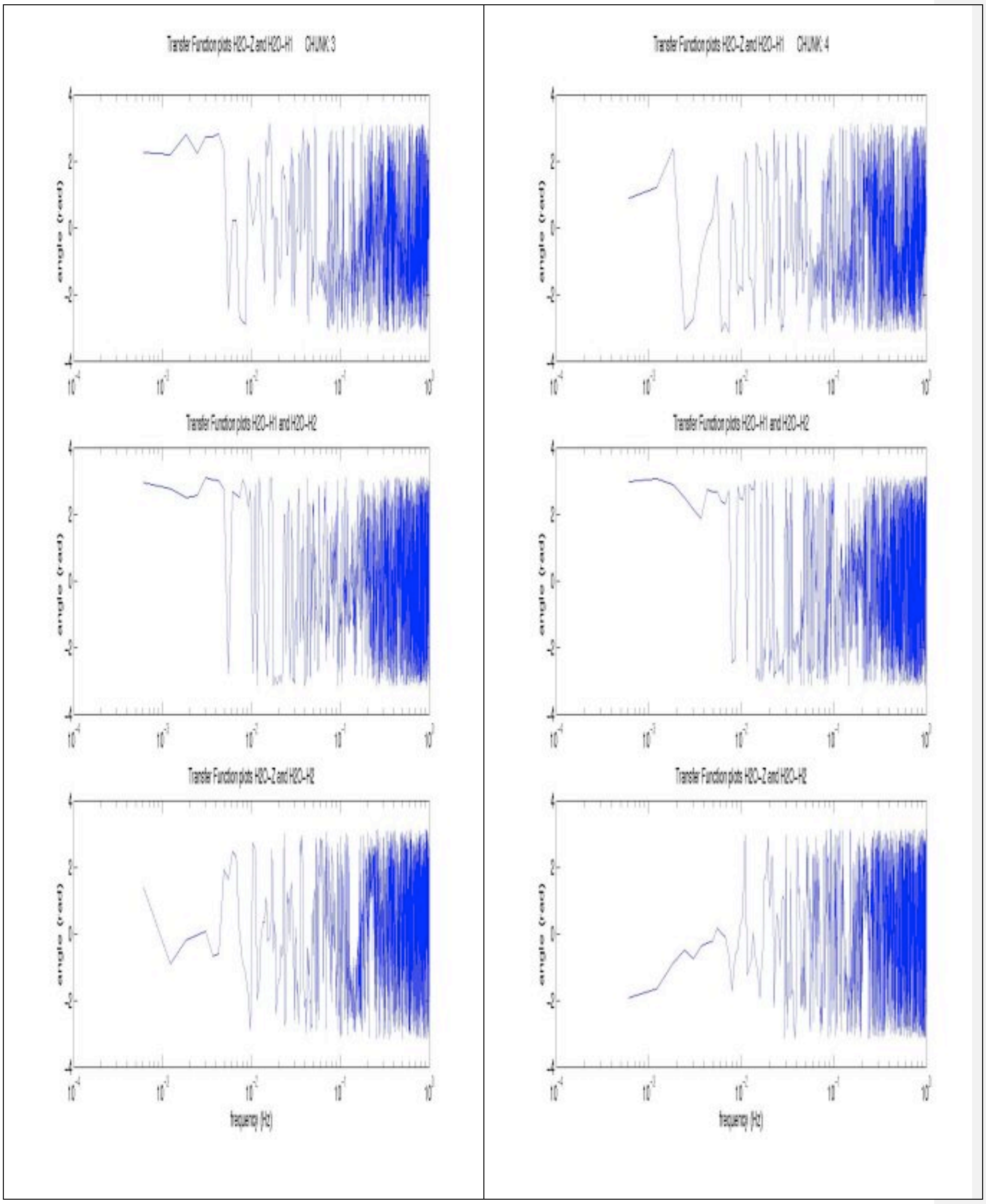


Fig. T37: Plot of the angle / phase of the transfer factions (calculated by using the theory of Crawford & Webb – using CSD and PSD to determine coherence) between the seismic channels. **H2O** data, day 121, buried seismometer. Chunks 3 and 4.

- **t17s_OSNtransfa.m**

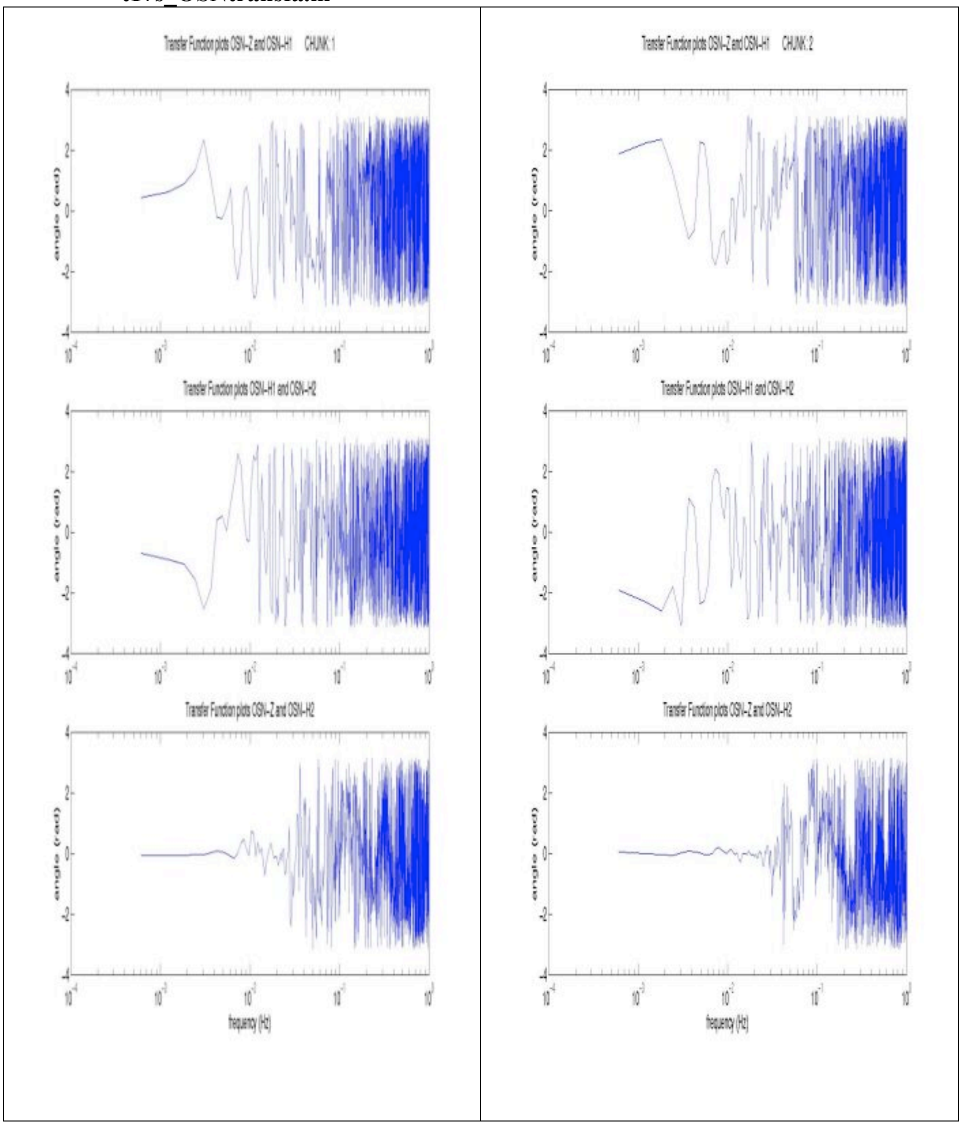


Fig. T38: Plot of the angle / phase of the transfer factions (calculated by using the theory of Crawford & Webb – using CSD and PSD to determine coherence) between the seismic channels. **OSN** data, day 62, borehole seismometer. Chunks 1 and 2.

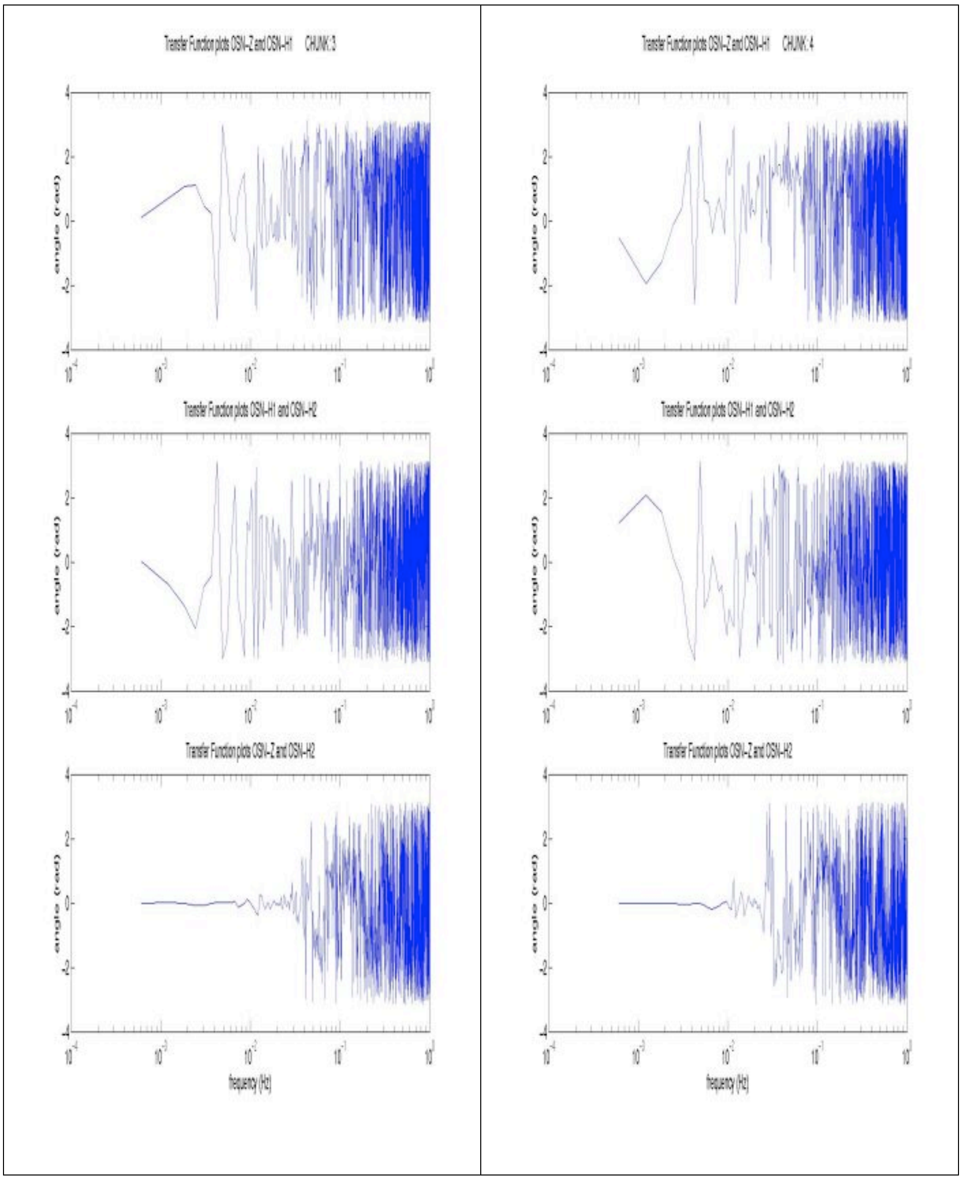


Fig. T39: Plot of the angle / phase of the transfer factions (calculated by using the theory of Crawford & Webb – using CSD and PSD to determine coherence) between the seismic channels. **OSN** data, day 62, borehole seismometer. Chunks 3 and 4.

- **t19_totalcorrectOSN.m**

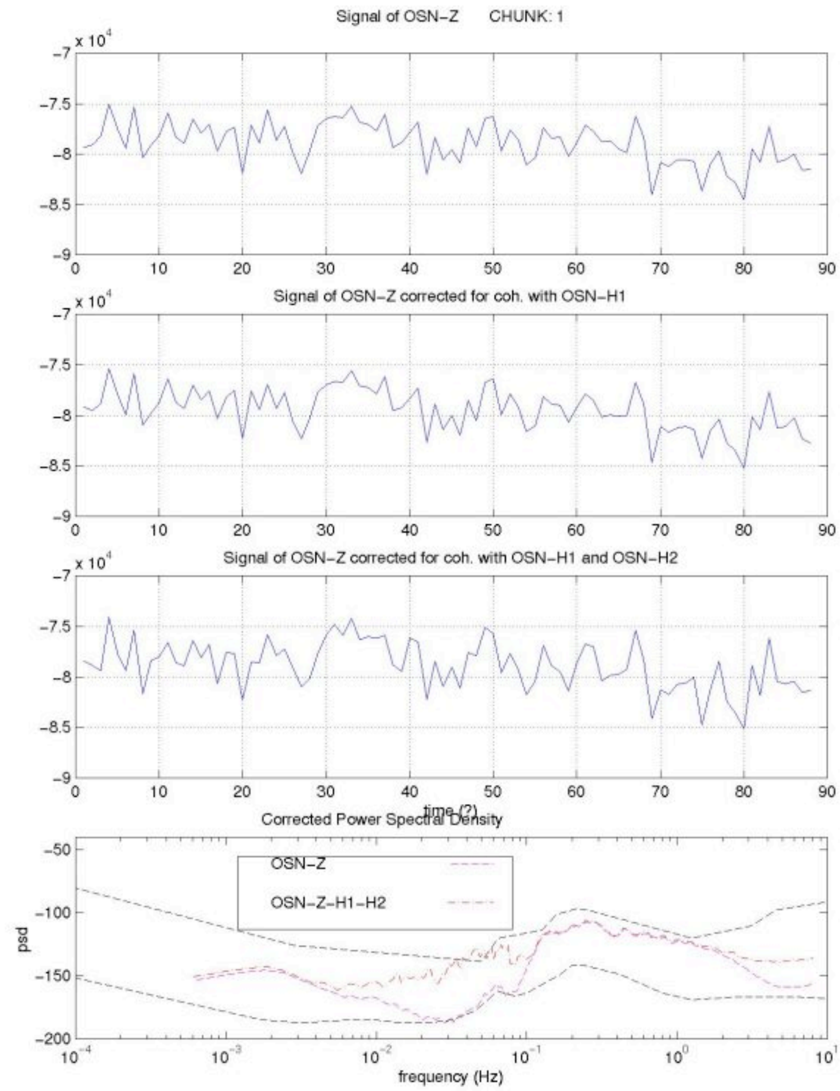


Fig. T40: Plot of attempt to calculate the corrected PSD's. Z and H2 (the original plots without correction); Z' and H2' (the plots corrected for coherent noise with H1); Z" (Z' corrected for coherent noise with H2'). But the corrected channels have higher noise levels than the originals.
OSN data, day 62, borehole seismometer. Chunk 1.

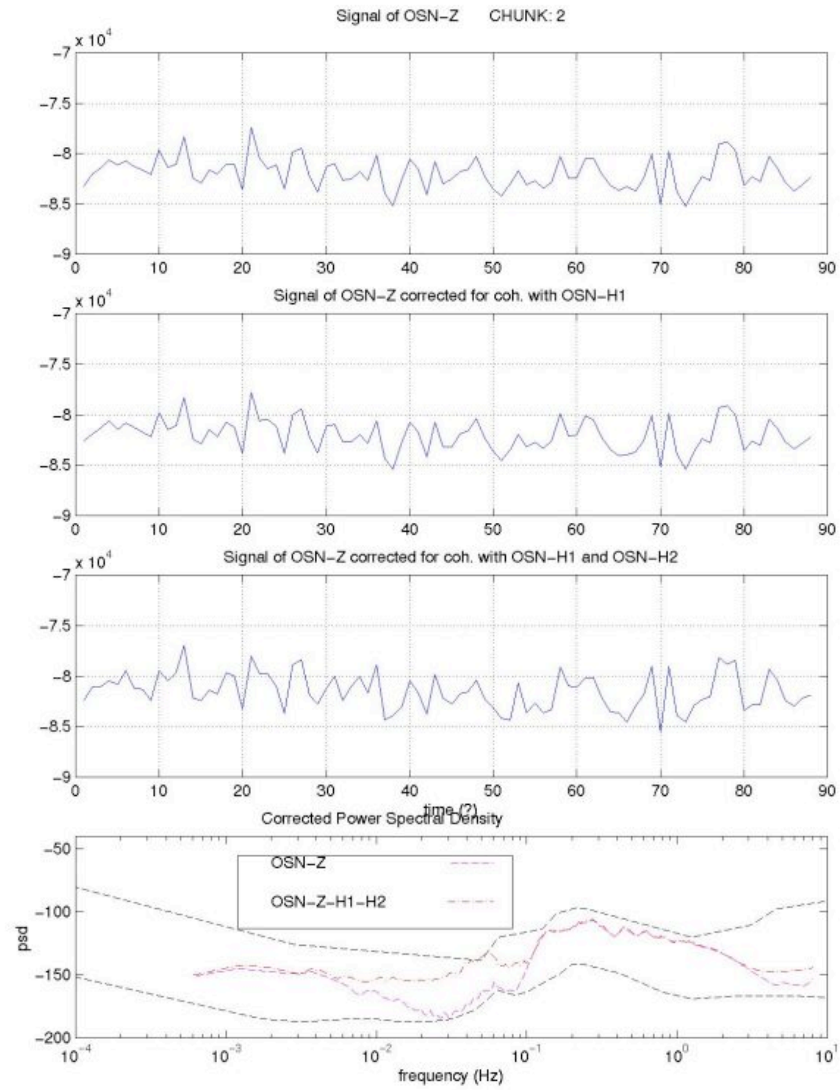


Fig. T41: Plot of attempt to calculate the corrected PSD's. Z and H2 (the original plots without correction); Z' and H2' (the plots corrected for coherent noise with H1); Z" (Z' corrected for coherent noise with H2'). But the corrected channels have higher noise levels than the originals.
OSN data, day 62, borehole seismometer. Chunk 2.

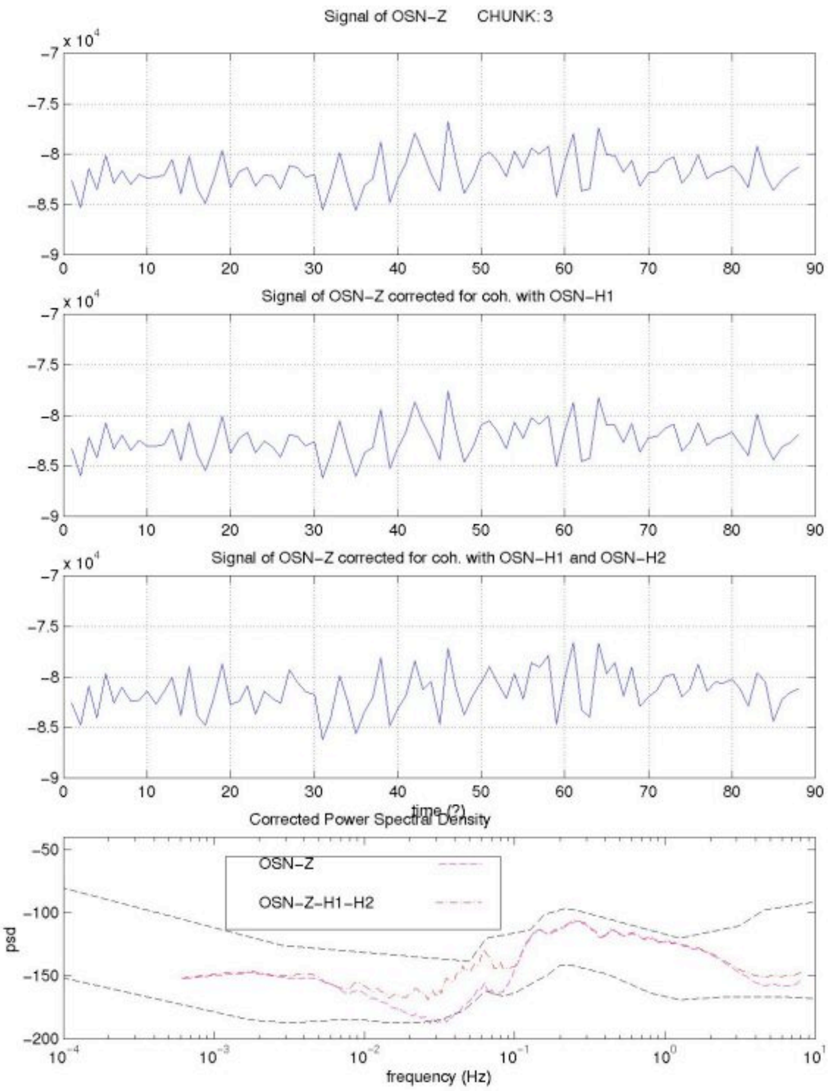


Fig. T42: Plot of attempt to calculate the corrected PSD's. Z and H2 (the original plots without correction); Z' and H2' (the plots corrected for coherent noise with H1); Z'' (Z' corrected for coherent noise with H2'). But the corrected channels have higher noise levels than the originals.
OSN data, day 62, borehole seismometer. Chunk 3.

- **t21_OSNwindcorZ.m**

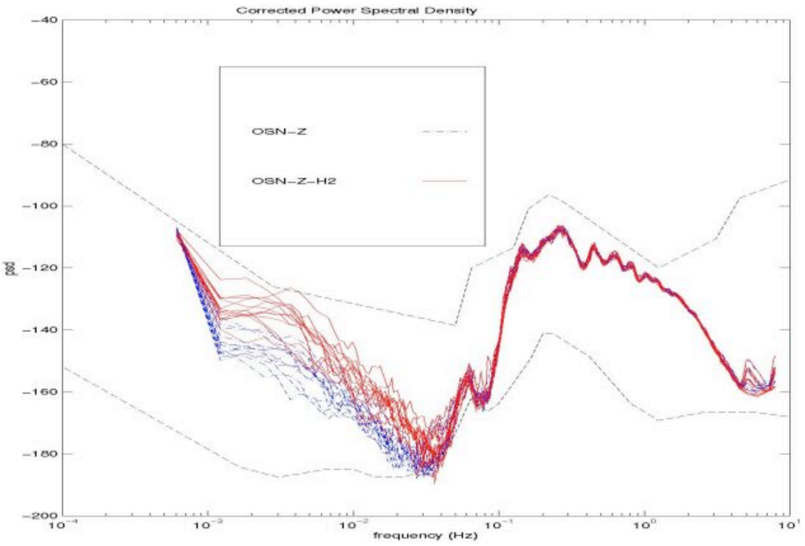


Fig. 43: Plot of attempt to calculate the corrected PSD's. Z (the original plot of Z without correction) and Z' (the plot corrected for coherent noise with H2). This time, to calculate the PSD and apply the transferfunctions to it, we don't use the MATLAB psd function. Instead we take this function to pieces (described also in 20_testnorm.m) and apply the transferfunctions to overlapping windows.
OSN data, day 62, borehole seismometer. All chunks.

Discussion and Conclusions

7. *What does the long-period psd spectrum look like?*

- t02s_tomgetspecdata_template1.m
- t03s_irenetest.m
- t04_puremfile.m
- t05_commentfile.m
- t07_threechan.m
- t08s_threechan.m

The **OSN** borehole seismometer long-period psd spectrum shows:

✓ The vertical channel is approximately 25 dB quieter than the horizontal in the infragravity wave band.

✓ The vertical is between 30 and 5 dB quieter than the horizontal in the noise notch. On the vertical the single-frequency peak is clearly visible, this is associated with direct transfer of ocean wave energy into elastic waves through non-linear coupling of waves and bathymetry [*Hasselman*, 1963].

✓ The noise floor (-185 dB) for the vertical channel lies at the transition between the infragravity band and the noise notch (0.03Hz). It is as quiet as the quietest continental sites.

✓ Then noise roof (-145 dB) for the vertical channel lies in the infragravity band (0.002 Hz).

✓ The noise floor (-165 dB) for the horizontal channels lies in the infragravity band (0.02 Hz).

✓ The noise roof (-120 dB) for the horizontal channel lies in the infragravity band (0.0015 Hz).

✓ Over one day of data the psd's of either channel show a spread of approximately 12 dB.

8. *How important is the "unavoidable" noise source, seafloor compliance?*

✓ If there is much seafloor compliance, there is usually a peak in the vertical component spectra at about 0.02 Hz. This peak is hardly visible to non-existent in the **OSN** borehole spectra. Because of the sensor being buried in a borehole compliance is expected to be low(er). When we look at the spectra of the buried and seafloor seismometers (Fig. A16), there is a clearly visible compliance peak.

9. How important is the "extraneous" noise source, seafloor currents (resulting in tilt noise)?

✓ Because this seismometer is buried, compliance-noise is reduced and therefore tilt-noise becomes more important. There might be a difference between tilt-noise on a seafloor seismometer or a borehole seismometer. But it is clear that circulation of water in the borehole can cause significant tilting of the instrument package.

✓ The horizontal and vertical spectra have similar slopes and are coherent, with much larger noise levels on the horizontal components. This indicates tilt-noise.

✓ Unfortunately I wasn't able to look at pressure data to confirm if noise levels varied in sync with ocean tides.

10. *What does the low-frequency coherence spectrum look like?*

- t09_OSNcohere.m
- t10_OSNcsd.m
- t10_H2Ocsd.m
- t11_OSNpsd+coh.m
- t12_H2Opsd+coh.m

✓ The coherence plots show that the only coherence worth mentioning is between the vertical (Z) and the horizontal (H2) component of the **OSN** seismometer.

✓ This coherence is close to 1.0 for frequencies up to 0.01 Hz. This is the area where tilt-noise dominates the signal, and it is rotated from H2 into the vertical component (Z).

✓ For frequencies from 0.01 Hz upward, first seafloor compliance starts to play a role. This results in less coherence around 0.01-0.04 Hz (the infragravity wave peak) and around 0.07 dB (the small compliance peak). At frequencies above 0.1 Hz seismic channels are dominated by microseism energy, so they are all partly coherent with one another.

✓ As expected there is no significant continuous coherence for the **H2O** buried seismometer. Burial is totally effective to cancel out tilt noise. It's worth noting that there could've been another coherent noise source, but there is not.

11. *What does the transfer function look like?*

- t13_OSNtransf.m
- t14s_H2Otransf.m
- t17s_OSNtransfa.m
- t16_H2Otransfa.m

✓ The transfer function between the vertical (Z) channel and the horizontal (H2) channel of the **OSN** borehole seismometer is roughly constant below 0.01 Hz where tilt dominates the vertical spectrum, and noisy and poorly resolved at higher frequencies. This is best shown by plots T38 and T39.

✓ It is not possible to determine the near constant values of the transfer function between Z & H2 because the plot is not detailed enough (these values will likely be in the order of 10^{-3} radians of tilt.

✓ The transfer function between horizontal components is a measure of the interaction of the two components due to flow noise and it varies between -3 and 3 radians. It is probably a complicated function of the site response and the noise source.

✓ For the **H2O** buried seismometer data, the transferfunction - in the plots where coherence is calculated by matlab 'cohere', Fig.T34 and T35 - is most regular between the two horizontal channels (H1 and H2). In the plots where the transfer function is calculated by using the theory of Crawford & Webb, there is no regularity in any of the transfer functions. It does vary less in the long-period part of the spectrum than in the high period part. This is a result of the low frequency of these movements.

12. *What is the difference between the vertical psd spectra before and after removing the coherent horizontal energy?*

- t19_totalcorrectOSN.m
- t21_OSNwindcorZ.m

✓ Figures T40 t/m T41 show the corrected signals in the time-domain and in the frequency domain for the OSN borehole seismometer. In *the time domain* there is not much difference between Z (the original vertical signal) and Z' (Z corrected for coherence with H1). Z" (corrected for H1 and H2) seems to have more noise, the signal is different.

✓ In *the frequency domain* the corrected signal clearly has higher noise levels in the band between 0.1 and 0.01Hz where seafloor compliance is important (infragravity wave peaks). In the band below 0.01 Hz, where tilt noise usually dominates, there is no improvement in the signal and in some places it is slightly noisier than the original signal.

Can the quality of the vertical channel seismic data be improved by subtracting the coherent horizontal energy (which is assumed to represent tilt-noise) ???

I can not answer this question as yet for the OSN borehole seismometer. I have had insufficient data (lacking pressure data and gravimeter data) and time. There must be a mistake in the M-file that we used to calculate the transfer functions. But the coherence graphs do show strong coherence between H2 and Z in the lower part of the long-period band of the spectrum. And in the spectrum of this seismometer compliance noise is limited. Therefore I conclude that if the M-file for calculating the transfer function and the corrected signal is revised, it is likely that the vertical component signal can be improved by subtracting coherent horizontal noise.

Part II

Studying ocean seismo-acoustic noise resonance's

Introduction

Shear waves in marine sediments have proved to be a significant factor in bottom-interacting ocean acoustics in some environments [*Hovem et al.,* 1991]. Specifically, it is shown that shear wave resonance's in a thin sediment layer over a rock substrate have a profound effect on acoustic reflection loss and shallow water acoustic transmission loss [*Hughes et al*., 1990].

Measurements of infrasonic seismo-acoustic noise using an OBS in shallow water have uncovered an unusual phenomenon: the noise spectrum of the horizontal component of seabed velocity shows several prominent peaks in the frequency range 0-8 Hz, whereas the noise spectra of both the acoustic pressure and the vertical component of seabed velocity show very weak or non-existent features at the same frequencies. Similar peaks have also been found with different OBS's and at different sites.

These features are interpreted theoretically by *Oleg A. Godin and David M. F. Chapman* in their paper *"Shear-speed gradients and ocean seismo-acoustic noise resonance's*" as **resonance's of vertical polarised shear waves in the upper sediment layer**, exited by the diffuse infrasonic sound field in water. Independent interface wave dispersion studies have revealed an approximate power-law profile of shear speed versus depth, having the form C(z)=Co*z^nu, with Co=21.5 and nu = 0.60. The theoretical development concentrates on

- exact analytic solutions for the resonance frequencies & wave field for power-law profiles
- the WKB and more advanced asymptotic solutions in the more general case of smooth shear-speed profiles with a power-law singularity.

Prominent peaks resembling those studied in the above paper are also recognised in the spectra of OSN and H2O sites. I've attempted to apply the theory of Godin & Chapman to these new broadband seismometer data, in order to constrain sediment thickness at the H2O site. Because this paper was the basis of my research, a short summary of their findings is given in the next chapter.

Scientific background

I : THE EXPERIMENT

What experimental data are presented?

The experimental site is located on the Scotian Shelf close to the Eastern Shore coastal region of Nova Scotia in a basin where the water depth is about twice as deep as the surrounding sand banks. The surficial sediment is composed of ponded layers of clay and silt over a harder material, probably glacial till.

The water depth is 156 m and the clay/silt layer is about 25.5 m thick. **The shear-speed profile can be viewed as an approximation to a continuous power-law profile of the form $c(z) = c_0 z^{\nu}$. The shear speed varies from 21.5 m/s (c_0) in the top 1 m to 150 m/s at the top of the till layer at 25.5 m depth. This profile agrees with the shear speed versus depth profile suggested by inversion of interface wave dispersion data.** Shear velocity in the hard sediment is estimated to be 900 – 1000 m/s.

The results are shown in Figure S1.

The resonance's in the noise on the horizontal geophone are clearly visible:

The first five peaks are at 1.06, 2.16, 3.24, 4.64, and 5.70 Hz. The sequence closely approximates a harmonic progression with a fundamental frequency of 1.11 ± 0.04 Hz. The vertical geophone channel displays similar features, but they are much weaker, showing only the first three peaks.

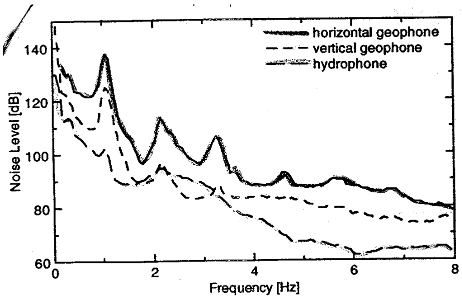


Fig. S1 Seismo-acoustic ambient noise resonance's measured using an ocean bottom seismometer in a shallow-water site on the Scotian shelf.

II : QUALITATIVE CONSIDERATIONS AND APPROACH

How can the noise resonance's be related to the normal modes of P-SV waves (Vertically Polarised Shearwaves)? What are the small parameters of the problem?

To explain the observed features, the following assumptions are made:

ass.1 Concerning the origin of the resonating sound waves, **it is assumed the water column supports a diffuse infrasonic noise field which interacts with the elastic seabed, generating vertically polarised shear waves (SV waves).**
Previous analysis of similar problems suggests that the strongest generation of shear wave would occur at the large impedance contrast at the sediment/substrate boundary [*Chapman et al.*, 1993]. Subsequent multiple transits of the sediment layer by the shear waves give rise to resonance's at frequencies that favour constructive interference.
Shear-wave resonance's are natural modes of vibration of this environment, their mode frequency & shapes are determined by seismo-acoustic propagation within the layered sediment.

ass.2 Density (ρ) is taken to be a constant.

ass.3 Concerning the angle of incidence of the rays and the resulting particle displacement, Snell's Law states that for very slow shear speeds the sediment-borne SV waves excited by water-borne sources would travel in an almost vertical direction, insensitive to the incidence angle of the wave in water. **Being transverse waves, these create almost-horizontal particle displacement.** In addition, the shear-wave particle motion at the water/sediment boundary is the sum of an upward-travelling incident shear wave and a downward travelling reflected shear wave. For low shear speeds, the shear wave is almost perfectly reflected with a phase inversion. This gives rise to a net displacement that is large in the horizontal direction and further reduced in the vertical direction. This is consistent with absence of resonance in the hydrophone and only a weak resonance on the vertical geophones.
Because of very small shear speeds, modal frequencies are independent of the angle of incidence of the exiting wave. The resonance frequencies can be determined by considering the normal-incidence case, reducing the problem to a well-known scalar wave equation.

ass.4 Concerning the bottom reflection coefficient, it would hardly be affected by the very low shear speeds, so the hydrophone is not expected to register large resonance effects.

ass.5 Concerning the bottom geometry and elastic parameters the assumption of a "layer-stacked model" is abandoned, since it was found to "present no expression that yields insight into the effects being studied" [*Hall et al*., 1995]. Instead, the **upper portion of the sediment is considered as a continuously stratified solid**. The shear modulus of marine sediments is small in the vicinity of the water-sediment interface and may vanish at the interface. Then it is natural to **assume a power-law dependence**

$$\mu = \rho_0 c_0^2 \ (z + z_1)^{2\nu}, \ \ \nu>0, \ \ z_1 \geq 0 \qquad (1)$$

of shear modulus (μ) on depth in the upper portion of the sediment column. Here ρ_0 is a constant with the dimensions of density ρ and c_0 = constant (shear-speed at the top of the sediment layer). There is both theoretical and empirical evidence in support for this assumption [*Hall et al.,* 1995, *Osler et al.,* 1996*, Hamilton.,*1976*, White,* 1983, *Iwasaki et al.,* 1997].
Figure S2 illustrates the model, and depicts the simple power-law profile ($z_1 = 0$) as well as a truncated power-law profile ($z_1 > 0$) formed by offsetting the singularity of the profile.

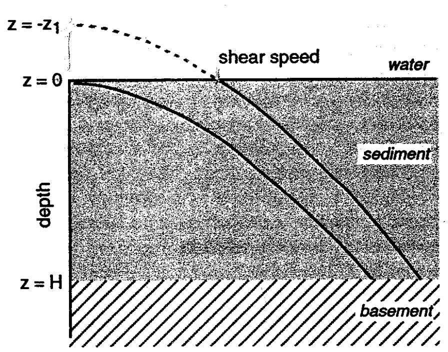


Fig. S2 Simple and truncated power-law shear-speed profiles in the sediment layer.

Gradients in P- and S-wave velocities beneath the seafloor are produced by loading (dewatering and compaction), diagenesis and lithological changes. S-wave velocities help us to understand the partitioning of seismic energy in the subsurface, anisotropy, energy transmission in surface waves, long range acoustic propagation in the water column. Determination of S-wave velocity is a means of detecting anomalous pore pressures because it is strongly dependent on effective stress. S-wave velocity can also be used to detect the dynamic shear modulus, which is an important parameter in evaluating bearing capacity of the deep sea floor and stability of submarine slopes. [Marine Geophysics]

Mathematically, an analytic treatment of the problem is made possible by the existence of two small parameters:

- **Δ_1**, the *ratio of Lame constants μ and λ* in the soft sediment.
  At the experimental site it is < 0.007.
- **Δ_2**, the *ratio of the horizontal component of wave vector to the shear-wave number* ($k = 2\pi/\lambda$) in the soft sediment.
  At the experimental site upper boundaries are 0.09 and 0.14 for ambient noise sources in the water column and deep in the bottom respectively.
- **T**, related to **Δ_2**, the *product of density and shear velocity ratios* in soft and hard sediments at their interface.
  At the experimental site it is on the order of 0.1.

At normal incidence ($\Delta_2 = T = 0$), the interfaces with the water and the hard sediment act like a free and rigid boundary for shear waves in soft sediment, and there exist normal modes of shear waves with real frequencies f_n.
At oblique incidence and/or when T is finite, the eigenfrequencies take on complex values because of radiation losses into water and the hard sediment. However, variations in the eigenfrequencies compared to the ideal case are small, including their imaginary parts, as long as Δ_2 and T are small. In the problem of broadband wave reflection from the sediment layer, the existence of complex eigenfrequencies near the real axis leads to frequency resonance's.

The horizontal displacement in S waves at the water/sediment interface ($u(0,f)$) tends to infinity when the frequency (f) tends to the eigenfrequency (f_n) and resonance peaks in horizontal displacement are related to normal modes of shear waves. In reality, the height of the resonance peaks is large but limited because of the small shift of eigenfrequency away from the real axis. So to determine the positions of the resonancefrequencies it is sufficient to study the eigenfrequencies of P-SV waves in the soft sediment layer.

III : THE EFFECT OF THE ANGLE OF INCIDENCE ON THE RESONANCE FREQUENCIES

Does analysis of the normal modes at normal incidence provide sufficient accuracy to determine resonance frequencies?

The relative variation in the frequency of the eigenmode due to variation of the angle of incidence is of the order of the square of the small parameter Δ_2. Therefore, under the environmental conditions at the experimental site, the spread of eigenfrequencies of a given normal mode due to various possible (real) incidence angles results in broadening of the respective spectral line in the power spectrum by a few percent at most. This is much less than the width of spectral lines observed in the seismo-acoustic ambient noise. So to determine eigenfrequencies it is sufficient to consider the case of the normal incidence only. At normal incidence SV and SH waves become indistinguishable.

IV : EXACT SOLUTIONS FOR THE POWER-LAW PROFILE

What is the exact solution for shear wave normal modes in a layer with a power-law profile of rigidity?

A. General solution

Considering elastic waves in a *continuously stratified solid* with *constant density* and a *power-law profile of the shear modulus* with $z_1 = 0$.

At *normal incidence* and $\upsilon \neq 1$, the exact solution of the wave equation for shear waves is given by:

$$\mathbf{u} = z^{1/2-\nu}\left[B_1 J_m\left(\frac{\omega}{c_0(1-\upsilon)} z^{1-\nu}\right) + B_2 Y_m\left(\frac{\omega}{c_0(1-\upsilon)} z^{1-\nu}\right)\right] \qquad (2)$$

and

$$\mathbf{m} = \frac{2\nu - 1}{2(1-\nu)}$$

u = displacement
z = depth
ν = exponential parameter of power-law profile
$\left.\begin{array}{l}\mathbf{J_m}\\ \mathbf{Y_m}\end{array}\right\}$ Bessel functions of the first and second kind
ω = frequency corresponding to two normal modes
m = a monotonically increasing function of ν, it varies between –1/2 and plus infinity when ν increases from 0 to 1.

I. Zeldenrust
Comment [1]: Nog uitzoeken wat deze parameter betekend, misschien was het een tiep fout en is het gewoon ν !

B. A layer with ideal boundaries

Considering a sediment layer $0 < z < H$ with *constant density*, a *power-law profile for the shear modulus*, $z_1 = 0$, and the following boundary conditions (BCs) imposed on the layer boundaries:

(stress tensor)	$\sigma_{13} = 0$	**(z = 0)**	(boundary with the inviscid fluid)	(3)
(displacement)	$u_1 = 0$	**(z = H)**	(rigid boundary)	(4)

This gives the resonance condition:

$$J_m\left(\frac{\omega}{c_0(1-\nu)} H^{1-\nu}\right) = 0 \qquad (5)$$

Leading to the expression:

$$f_n = (2\pi)^{-1} c_0 (1-\nu) H^{\nu-1} j_{m,n} \qquad (6)$$

for the *n*th resonance frequency, $n = 1,2,\ldots$

All the resonance frequencies are real. It follows from the properties of the Bessel function zeros that for a given ν the sequence $\{f_n\}$ is not generally equally spaced: $f_{n+1} - f_n \neq f_n - f_{n-1}$, but rapidly becomes so as *n* increases.

The sequence of the resonance frequencies can only be spaced equally if $m^2 = ¼$. That is, among the power-law profiles (other than the trivial case of μ constant) only the one with ν = 2/3 can possibly result in an equi-spaced set of resonance frequencies.

C. Truncated power-law profile

Considering now a somewhat more general geoacoustic model. The *assumptions from section B are retained*, but *z_1 is taken to be positive in Eq. (1)*. This model is referred to as the truncated power-law profile. Of primary interest is the effect of the small but nonzero μ (0) has on the resonance frequencies. The general solution of the wave equation is still given by Eq. (2) if $z + z_1$ is substituted for z in the right side of that equation. BCs Eqs. (3) and (4) can now be written as

$$B_1 J_{m+1}(a) + B_2 Y_{m+1}(a) = 0 \qquad a = \omega z_1^{1-\nu} / c_0(1-\nu) \qquad (7)$$
$$B_1 J_m(b) + B_2 Y_m(b) = 0 \qquad b = \omega (H + z_1)^{1-\nu} / c_0(1-\nu) \qquad (8)$$

By requiring that these simultaneous equations have a nontrivial solution with respect to amplitude constants $B_{1,2}$, the resonance condition becomes

$$J_{m+1}(a)\, Y_m(b) - J_m(b)\, Y_{m+1}(a) = 0 \qquad (9)$$

D. The effect of the boundary conditions

In the sections B and C, ideal boundary conditions have been assumed. While the first BC is the correct BC for shear waves at normal incidence on a solid/fluid interface – provided fluid viscosity and surface roughness are negligible – the second BC corresponds to an absolutely rigid surface and is a strong idealisation. Instead of imposing that BC at the boundary z = H, It is now assumed that half-space z > H is occupied by a homogeneous solid with density ρ_2 and shear wave velocity c_2. At a solid/solid interface with a "welded" contact, shear waves at normal incidence are not coupled to compressional waves. The physical requirements of displacement and traction continuity at the interface reduce to two boundary conditions, namely, continuity of the x-component of the displacement and of the σ_{13} component of the stress tensor. This leads to the following BC for waves within the sediment layer:

$$u_1 + i\omega^{-1} cT\, \delta u_1/\delta z = 0$$

$$(z{=}H), \text{ where } T = \rho(H)\, c(H)/\rho_2 c_2 \qquad \text{(Hooke's law)} \qquad (10)$$

where ρ(H) and c(H) are the density and shear-wave velocity in the layer just above the boundary with the solid half-space. T is the only parameter characterising the effect of the substrate on the normal modes of the shear waves. In the limit T → o Eq. (10) turns into Eq. (4). That is why the presence of the substrate with finite impedance can be considered as a perturbation with respect to the idealised problem with a rigid boundary.
The resonance condition than becomes:

$$J_m(\omega\tau) = i\, T\, J_{m+1}(\omega\tau), \quad \tau = H^{\nu-1} / c_0(1-\nu) \qquad (11)$$

For the resonance frequencies within *a layer between fluid and solid halfspaces* the following condition can be derived:

$$f_n = \frac{c_0(1-\nu)}{2\pi} H^{\nu-1} \left(j_{m,n} - iT + \frac{1+2m}{2 j_{m,n}} T^2 + O(T^3) \right) \qquad (12)$$

Penetrability of the boundary in the realistic model results in perturbations of both real and imaginary part of f_n. Because of the wave energy leakage through the boundary, the resonance frequencies cease to be real-valued. With broadband excitation, the complexity of f_n manifests itself in the power spectrum as smearing of spectral lines, with the width of the lines (approximately) independent of their frequency. This spectral line broadening is the only major effect of the finite impedance of the lower boundary of the nontruncated layer.

Considering *a truncated layer above a solid halfspace* the BC at the upper boundary leads to Eq. (7), while the BC Eq. (10) becomes

$$[J_m(b) - iT\, J_{m+1}(b)]\, B_1 + [Y_m(b) - iT\, Y_{m+1}(b)]\, B_2 = 0, \qquad (13)$$

$$b = \frac{\omega}{c_0(1-\nu)} (H + z_1)^{1-\nu}$$

There is one special case, which is of primary interest. When $n = O(1)$, $a << 1$, $T << 1$, for the resonance frequencies the following condition can be derived:

$$f_n \approx \frac{c_0(1-\nu)}{2\pi} (z_1 + H)^{\nu-1} j_{m,n} \left(1 + d_{m,n} \frac{z_1}{z_1 + H} - \frac{iT}{j_{m,n}}\right) \qquad (14)$$

This equation describes perturbations in the resonance frequencies due to
- Finite impedance of the lower boundary
- Small but nonzero shear-wave velocity nears the upper boundary.

To leading order, the perturbations are additive as it should have been expected. These two perturbations have already been discussed above.

Theoretically, the continuous variation of shear-speed (power law) introduces a term involving the logarithmic derivative of shear modulus into the wave-equation. The combined effects of this term and the upper boundary conditions remain significant even away from the interface. The net effect is a ν-dependent frequency shift of the resonances.

V : ASYMPTOTIC SOLUTIONS OF THE SHEAR-WAVE EQUATION

What are the effects of deviations from the power-law on the exact solution for shear wave normal modes in a layer?

What are the assumed deviations?
- Variable density
- Shear modulus profiles deviating from the power-law depth dependence.

In this section *WKB* [*Brekhovskikh and Godin*, 1990/1998] and more advanced *asymptotic approximations* to solutions of the shear-wave equation at normal incidence are considered.

A. WKB approximation and it's conditions of use

The existence of shear-speed singularity (in the vicinity of the surface, shear-speed vanishes in a singular manner) in the sediment invalidates the WKB method for this problem. When $0<\nu<1$, $\in \rightarrow \infty$ and the error of the WKB approximation increases indefinitely when $z \rightarrow 0$ making the approximation meaningless. The **only exception occurs when ν=2/3**. Then $\in$ is identically zero, and the WKB solution is not just a good approximation but it is *precisely equal* to the exact solution. It can be concluded that the power-law profile with ν=2/3 is the *only case* of a continuously stratified, inhomogeneous solid of constant density when WKB solutions are exact for shear waves at normal incidence.

B. Uniform asymptotics

The WKB method enables one to derive high-frequency asymptotics of solutions to the wave equation with an *arbitrary* smooth depth dependence of medium parameters , but with a few exeptions is not applicable in a vicinity of the horizon where the shear modulus vanishes. The exact solutions have no limitations on frequency or depth but refer to a media with ρ = const and specific μ (z). **The uniform asymptotic approximation (= corrected WKB approximation) in this section posesses the virtues of both the previous approaches.**

- **Like the WKB approximation, it applies general smooth variation of medium parameters**
- **Like the exact solution, remains valid when shear-modulus vanishes according to the power law**

To derive a wave field asymptotics applicable at z → 0, the *reference equation method* [*Brekhovskikh and Godin*, 1990/1998] is applied. The resonance condition for shear waves in a layer 0 < z <H between two homogeneous fluid and solid half-spaces becomes

$$J_m(\omega\tau) = i\, T\, J_{m+1}(\omega\tau), \quad \tau = H^{\nu-1} / c_0(1-\nu) \qquad (11)$$

Imposing the BC Eq (10) at z=H one obtains the resonance
condition Eq.(11) where τ stands for the wave travel time between the boundaries z=0 and z=H. The only difference from the exactly solvable case involves the value of τ which now is not related to H and ν in a simple way as in Eq.(11).
Consider the resonance condition Eq.(11) in the case T=0 (rigid lower boundary). Then the resonance frequencies are $\omega_n = j_{m,n} / \tau$. Using the asymptotic expansion for the Bessel function zeros [*Abramowitz and Stegun*, 1964] one has at n >>1:

$$\omega_n = \frac{\pi}{\tau}\left(n + \frac{m(\nu)}{2} - \frac{1}{4}\right) + O\left(\frac{1}{n}\right) \qquad (15)$$

In the case of m = ½ Eq.(15) is exact for all n = 1,2,…

VI : ***DATA ANALYSIS AND COMPUTATIONAL EXAMPLES***

Can the analytically derived conclusions be illustrated with computational results?

a. the simple power-law profile (section IV-B)

The resonance frequencies are expressed in terms of the zeros of Bessel functions in Eq. (6):

$$f_n = (2\pi)^{-1} c_0 (1 - \nu) H^{\nu-1} j_{m,n}$$

For a shear wave at normal incidence, the TWTT through the upper sediment layer is
$2 H^{\nu-1} / c_0 (1 - \nu)$, so all frequencies can be normalized by the reference frequency:

$$F = c_0 (1 - \nu) / 2 H^{\nu - 1} \tag{14}$$

(F is the frequency spacing of adjacent resonances in the limit of large mode order, it is also the frequency of the fundamental resonance in the special case ν = 2/3.) Figure 10 shows the normalized shear resonance frequencies versusν for the first five modes, calculated three different ways:
(1) an asymtotic approximation, (2) exact solution and (3) WKB approximation (the resonance frequencies are simply the sequence of integers). Methods (2) and (3) have nearly identical results, the only difference occurring at low mode order. Note that the resonance frequencies are exactly harmonic in the special case ν=2/3. Note also that in the limit ν→0, which corresponds to a homogeneous layer, the sequence of normalized frequencies approaches *n* – ½. The precision of the asymptotic approximation Eq.(15) allows a straightforward reduction of the experimental data reported in Sec.I. To first order in 1/*n*, Eqs.(15) and (6) can be written as:

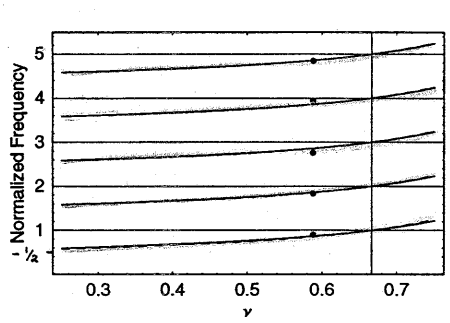


Fig. S3: Normalized resonance frequencies f_n/F vs exponent value ν in the shear-speed profile $c=c_0 z^{\nu}$ with a rigid bottom boundary for shear resonances 1-5. Solid lines: exact solution; dashed lines: asymptotic approximation Eq.(15); thin horizontal lines: WKB approximation; solid circles: observed resonances, normalized by F=1.18 Hz.

$$f_n = F(n + m/2 - 1/4) \tag{15}$$

which is applied to the sequence of five noise resonances to determine the free parameters by linear regression. The resulting fit gives F = 1.18 and *m* = 0.21, which corresponds to the shear speed profile parameters c_0=21.7±1.5 and ν=0.59±0.06. The experimental resonance frequencies, normalized by F = 1.18, have been added to Fig.10, plotted at the exponent value ν=0.59. The good agreement between experimental and modeled resonances supports the hypothesis that the observed noise peaks are shear-wave resonances in a sediment layer having a power-law shear-speed profile. (It should be emphasized, that two shear-speed profiles having the same TWTT and power-law singularities of the shear-speed at z→0 with the same ν couldn't be distinguished by their resonance frequencies.) Owing to the natural uncertainties in the experimental data, no further improvement in the agreement is expected by introducing complicating factors, such as a truncated profile or a non-ideal bottom boundary condition.

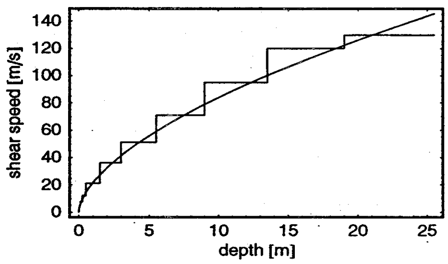


Fig. S4: The experimental shear-speed profile. The staircase is the stacked layer profile previously derived from interface wave dispersion and the continuous line is the power-law profile $c(z)=21.7\ z^{0.59}$ (SI units) implied by the noise resonances.

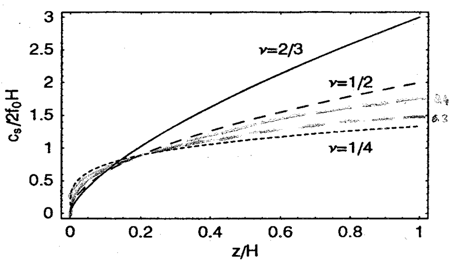


Fig.S5: Shear speed versus depth for power-law profiles having different exponential parameters, keeping fixed the TWTT through the layer.

The implied power-law shear-speed profile (Fig.S4) is nearly identical to that independently determined from analysis of interface wave dispersion at the same site [*Osler and Chapman*, 1996]. Figure S5 shows three power-law profiles having different exponents ν but the same TWTT, that is, they all have the same reference frequency F. For each profile, Figure S6 shows the shapes of the first three resonances in this ideal environment, assuming a perfectly rigid boundary at the bottom. The functions describing horizontal displacement are normalized to unity at the upper boundary, and are evaluated at the resonance frequencies given by Eq.(6). Note that at larger values of the exponent the modes are trapped near the surface, while at small values of the exponent the modes span the sediment layer. In the limit ν→0, the layer becomes isospeed and one would expect cosinusoidal modal shapes. The modes for the experimental case, having the exponent ν=0.59, are very similar to those for the case ν=2/3.

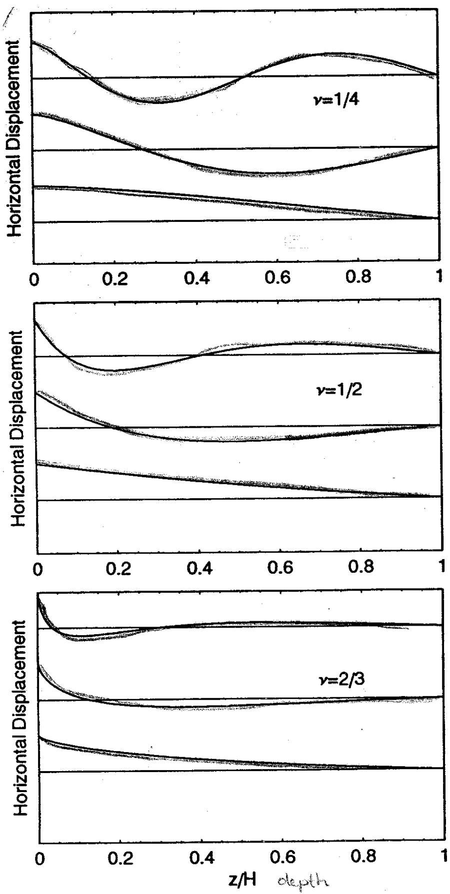


Fig. S6: Shapes of the first three shear-wave resonances associated with the profiles of Fig.12. The zeros of the functions are offset by their normalized resonance frequencies.

b. <u>the truncated power-law profile (section IV-C)</u>

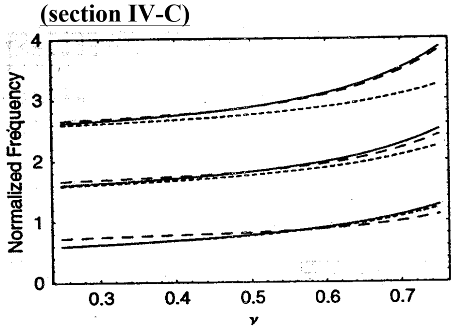


Fig. S7: Normalized resonance frequencies versus exponent value ν in the shear-speed profile c= c_0 (z + z_1)$^{\nu}$ with a rigid bottom boundary for shear resonances 1-3. Solid lines: exact solution; long dashed lines: the corrected WKB approximation; short dashed lines: the nontruncated solution.

The <u>resonance condition</u> is Eq.(9):

$$J_{m+1}(a)\, Y_m(b) - J_m(b)\, Y_{m+1}(a) = 0$$

Computed for the case z_1/H=0.01, the resonance frequencies given by (1) the exact solution and (2) the WKB approximation are compared in Fig.S7 alongside the results from the simple (nontruncated) profile having the same parameters. Note that the truncated profile gives higher frequencies, owing to the reduced travel time through the sediment. The WKB approximation improves with mode order.

So in the problem considered, the resonance frequencies are primarily determined by

- **TWTT of shear waves in the sediment layer**
- **The exponent ν, which governs the type of shear-speed singularity**

In practice, if only resonance frequencies are measured there is little hope to distinguish between two distinct geoacoustic models sharing the same values of these parameters. Also of some importance are:

- **Elastic parameters of the substrate**
- **Surface value of the shear modulus**

Experimental method

✓ ***MAIN QUESTION***:

What is the thickness of the sediments at the H2O site???

Related questions and actions:

13. What does the short-period psd spectrum look like?
    (Plot the short-period spectrum for several days of H2O data)
14. Where are the resonance peaks?
    (Pick all peaks in the PSD plots of OSN and H2O data)
15. What does this infer for the sediment thickness?
16. Are the resonance peaks constant throughout one day of data?
    (Plot the data of the H2O site in a colour spectrogram to determine how the (resonance) peaks vary over 24-hours)
17. Are the resonance peaks constant throughout several days of data?
    (Plot several days of H2O data in colour spectrograms to determine how the (resonance) peaks vary with different days)
6. Do peaks vary between Z, H1 and H2 channels and why?

✓ ***DATA***

I used 2 days (091 and 121) of H2O spectra:

/osn1/b3s2/OSN/H2O/work/H2O_spectra/day091_160.frequency
/osn1/b3s2/OSN/H2O/work/H2O_spectra/day091_160_%2.2d_Z.spectra
/osn1/b3s2/OSN/H2O/work/H2O_spectra/day091_160_%2.2d_H1.spectra
/osn1/b3s2/OSN/H2O/work/H2O_spectra/day091_160_%2.2d_H2.spectra
/osn1/b3s2/OSN/H2O/work/H2O_spectra/day121_160.frquency
/osn1/b3s2/OSN/H2O/work/H2O_spectra/day121_%2.2d_C1.spectra
/osn1/b3s2/OSN/H2O/work/H2O_spectra/day121_%2.2d_C2.spectra
/osn1/b3s2/OSN/H2O/work/H2O_spectra/day121_%2.2d_C3.spectra

✓ ***SOFTWARE***

I created MATLAB M-files[•] to do al kinds of calculations with the data. The following steps were taken:

- r01_FC160psdH2O***.m

This M-file plots the Power Spectral Density of 3 channels (Z, H1, H2) of a seismometer on sediments at the H2O site. It loops through 12/13/14 chunks of data into which days 091/100/121 are divided. The sampling frequency (FC) is 160samples/s. It plots the microseism and VLF part of the spectrum (so freq.0.1-100).

The data are then manipulated in the following steps:

14) The frequency data in the 'frequency'-file is opened.

15) Binary data are read from the file.

16) The file is closed.

For each of the 13 chunks (1.8 hour) of data:

17) PSD information for channel Z is read from a 'Z.spectr'-file

---

• · Appendix G

18) The PSD of Z is plotted with a logaritmic scale and the plot put on 'hold on'.
19) PSD information for channel H1 is read from a 'H1.spect'-file
20) The PSD of H1 is plotted in the same plot
21) PSD information for channel H2 is read from a 'H2.spect'-file
22) The PSD of H2 is plotted in the same plot
23) The plot is printed

- r02_FC160tomH2O***.m

This M-file plots the Power Spectral Density of 3 channels (Z, H1, H2) of a seismometer on sediments at the H2O site. The sampling frequency (FC) is 160samples/s. It plots the (microseism and) VLF part of the spectrum (so freq.0.9-60) with a linear or a logarithmic scale.

- r03_polyfit5H2O***.m
- r04_polyfit13H2O121.m

This M-file uses the command **'ginput'** to pick (resonance) peaks in a plot showing PSD of 1 chunk of H2O data of (Z,) H1, H2. After picking these points they are plotted in a graph using ordinal numbers and the best fit line is calculated by command **'polyfit'**. This line is a fit to the f_n and has a function $\mathbf{f_n = F\mathit{n} - b}$. Then the method described in the paper of *Godin an Chapman* (previous chapter)is used to determine the thickness of the sediments:

The Godin-Chapman Method

The Godin and Chapman method assumes a power-law profile for the shear speed in the sediments and it assumes that the sediments are underlain by a rigid boundary. The depth dependence of the shear modulus is given by:

$$\mu = \rho_0 c_0^2 z^{2\nu}, \nu > 0 \qquad (1)$$

where ρ_0 is a the density which is assumed constant, c_0 is the uppermost shear velocity, z is depth and ν gives the power law dependence. The two-way travel time (T) for a vertically propagating shear wave through the upper sediment layer is:

$$2H^{1-\nu} / c_0(1-\nu) \qquad (14)$$

where H is the thickness of the sediment. The normalized frequency (1/T)

$$F = c_0(1-\nu) / (2H^{1-\nu}) \qquad (14)$$

corresponds to the fundamental resonance when ν=2/3. The relationship between the resonant frequencies, f_n, and their ordinal number, n, is approximately linear:

$$f_n = F(n + m/2 - 1/4) \qquad (15)$$

Together with $f_n = Fn + b$ this gives:

$$m = \frac{1}{2} - 2b / F$$

where m is related to ν by:

$$\nu = (2m+1) / [2(m+1)] \qquad (2)$$

From the slope and intercept of the resonant frequencies one can compute the ratio: $c_0 / H^{(1-\nu)}$.
Then given either the sediment thickness or the uppermost shear velocity one can compute the other quantity.

freqZ	=an array with the frequencies of the first 5-13 frequency peaks in the PSD-plot of Z;
freqH1	= an array with the frequencies of the first 5-13 frequency peaks in the PSD-plot of H1;
freqH2	= an array with the frequencies of the first 5-13 frequency peaks in the PSD-plot of H2;
powerZ	= an array with the PSD's of the first 5-13 frequency peaks in the PSD-plot of Z;
powerH1	= an array with the PSD's of the first 5-13 frequency peaks in the PSD-plot of H1;
powerH2	= an array with the PSD's of the first 5-13 frequency peaks in the PSD-plot of H2;
fn	=resonance freq.;
n	=the ordinal numbers;
F	=the slope of the best fit line through the fn;
b	=the intersection with y-axes of this line;
Co	=the shear speed at top of the sediments;
H	=the thickness of the sediments;
m	=a monotonically increasing function of ν, varying between -0.5 and plus infinity;
ν (nu)	=power of the velocity-depth function

The data are then manipulated in the following steps:
1) **The frequency data in the 'frequency'-file is opened.**
2) **Binary data are read from the file.**
3) **The file is closed.**

For each of the 13 chunks (1.8 hour) of data:

4) **PSD information for channel Z is read from a 'Z.spectr'-file**
5) **The PSD of Z is plotted with a logaritmic scale and the plot put on 'hold on'.**
6) **The command** 'ginput' **is used to pick 5 or 13 (resonance) peaks of Z in the plot**
7) **PSD information for channel H1 is read from a 'H1.spect'-file**
8) **The PSD of H1 is plotted in the same plot**
9) **The command** 'ginput' **is used to pick 5 or 13 (resonance) peaks of H1 in the plot**
10) **PSD information for channel H2 is read from a 'H2.spect'-file**
11) **The PSD of H2 is plotted in the same plot**
12) **The command** 'ginput' **is used to pick 5 or 13 (resonance) peaks of H2 in the plot**
13) **The command** 'polyfit' **is used to calculate the best fit line to the peaks**
14) **The picked peaks and the line are plotted in a graph using ordinal numbers**
15) m **is calculated**
16) ν **is calculated**
17) **the** ratio **is calculated**

- r05_polyfitOSN.m

This M-file uses the command 'ginput' **to pick (resonance) peaks in a plot showing PSD of OSN data of (Z,) H1, H2 (buried seismometer). After picking these points they are plotted in a graph using ordinal numbers and the best fit line is calculated by command** 'polyfit'**. Then the method described in the paper of *Godin an Chapman* (previous chapter)is used to compute a typical uppermost shear velocity (C_0) for deep sea sediments because the sediment thickness at OSN-1 is known from drilling (242m).**

- r06_pickallrfreqH2O121.m

This M-file picks 30 (resonance) peaks in a plots showing PSD of 1 chunk at the time of H2O data from day 121 of H1 and H2. To pick I used the command **'ginput(30)'**.
The freq + power of these peaks were saved in the arrays *freqH1/H2* and *powerH1/H2*, the PSD's were also saved in the arrays *H1, H2* and *f*. Al these arrays were saved together in a file named *the30picks*.

- r07_plotallrfreqH2O121.m

This file loads and plots 30 picked (resonance) frequency peaks for each chunk (total of 13) of data from day 121 of the H2O site. These peaks were picked in psd plots in the band of 1Hz-70Hz and saved in M-file **r06_pickallrfreq.m** as *the30picks*. The (resonance) frequencies are plotted against PSD.

- r08_plotchunkrfreqH2O121.m

This file loads and plots 30 picked (resonance) frequency peaks for each chunk (total of 13) of data from day 121 of the H2O site. These peaks were picked in psd plots in the band of 1Hz-70Hz and saved in M-file **r06_pickallrfreq.m** as *the30picks*. The (resonance) frequencies are plotted against time, in chunks of 1.8 hours.

- r09_plot29rfreqdifH2O121.m

This file calculates and plots the differences between adjacent resonance's frequency peaks for all chunks (total of 13) of data from day 121 of the H2O site. These peaks were picked in psd plots in the band of 1Hz-70Hz and saved in M-file **r06_pickallrfreq.m** as *the30picks*.

- r10_trytick.m

Trying to get tickmarks in the plot. Using the Object property 'XTick', but it doesn't work (yet).

- r11_colorarrayH2O***.m

This M-file opens files containing psd's of H2O data from day 091/121/100. For each channel (Z, H1, H2), data are saved in a in arrays of 13 rows and 1576 column's called *powZ*, *powH1*, *powH2*. (The last column consists of only NaN's.) These values are saved in a files called *colorpl091.mat*, *colorpl121.mat* and *colorpl100.mat*. This is done for the purpose of creating colourplots in following M-files: **r12_colorplotH2O***.m** and **r13_logcolorplotH2O***.m**.

- r12_colorplotH2O***.m

This M-file creates a plot for data of H2O, day 091/121/100, with

- x axis = chunk number (time)
- y axis = frequency (in this case the high frq. from 0.9-60Hz), linear.
- colours = indicate amplitude of PSD

The arrays *powZ, powH1, powH2* were calculated in another M-file, **r11_colorarrayH2O***.m**.

- r13_logcolorplotH2O***.m

This M-file creates a logplot for data of H2O, day 121, with

- x axis = chunk number (time)
- y axis = frequency (in this case the high frq. from 0.9-60Hz), logarithmic
- colours = indicate amplitude of PSD

The arrays powZ, powH1, powH2 were calculated in another M-file, **r11_colorarrayH2O***.m**.

Results

- **`r01_FC160psdH2O***.m`**

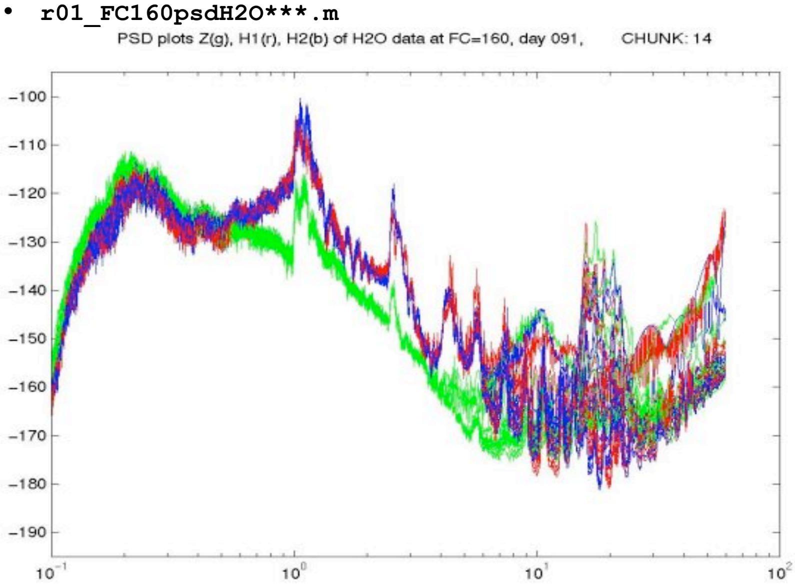


Fig. S8: PSD plot of vertical (Z) and horizontal (H1, H2) component spectra. **H2O** data, day 091, buried seismometer. All (14) chunks (1 chunk is approximately 1.7 hours of data).

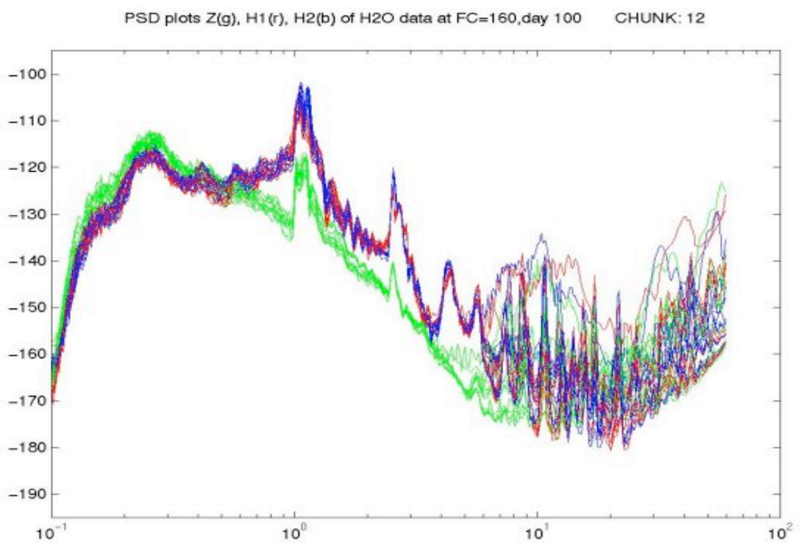


Fig. S9: PSD plot of vertical (Z) and horizontal (H1, H2) component spectra. **H2O** data, day 100, buried seismometer. All (12) chunks (1 chunk is approximately 2.0 hours of data).

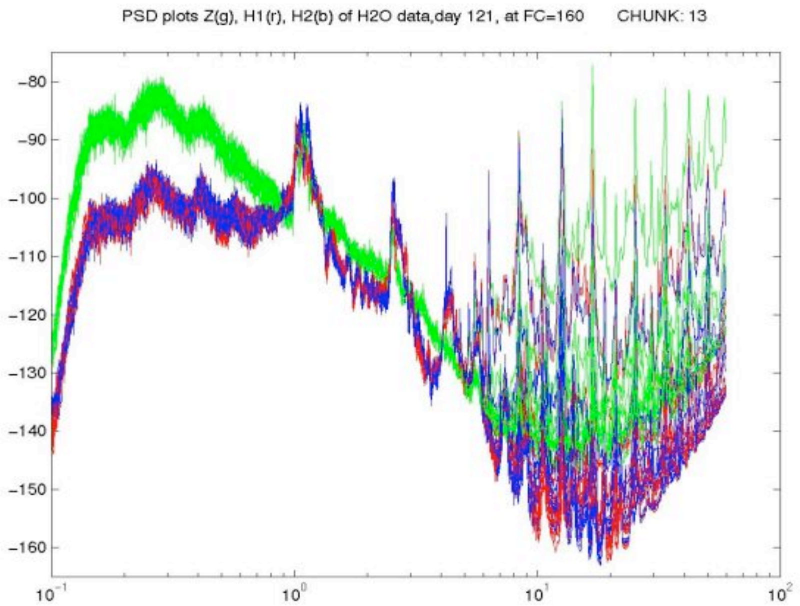


Fig. S10: PSD plot of vertical (Z) and horizontal (H1, H2) component spectra. **H2O** data, day 121, buried seismometer. All (13) chunks (1 chunk is approximately 1.8 hours of data).

- **`r02_FC160tomH2O***.m`**

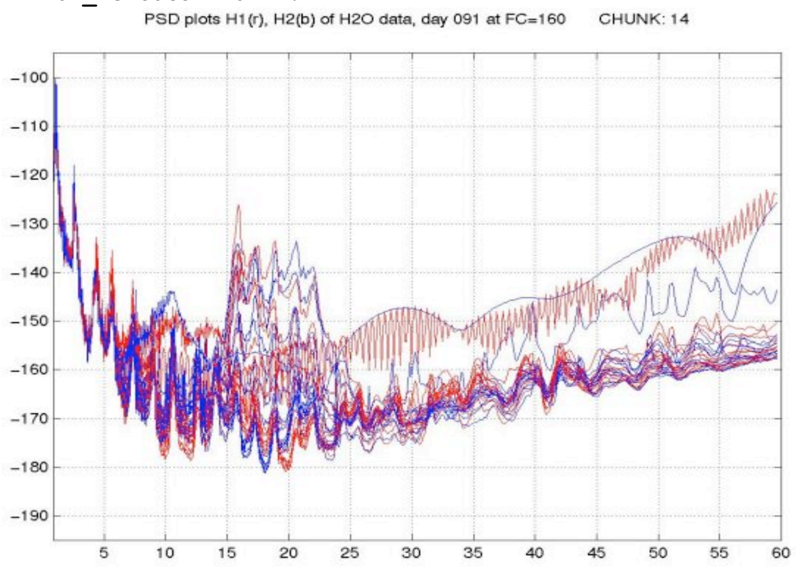


Fig. S11: linear PSD plot of vertical (Z) and horizontal (H1, H2) component spectra. **H2O** data, day 091, buried seismometer. All (14) chunks (1 chunk is approximately 1.7 hours of data).

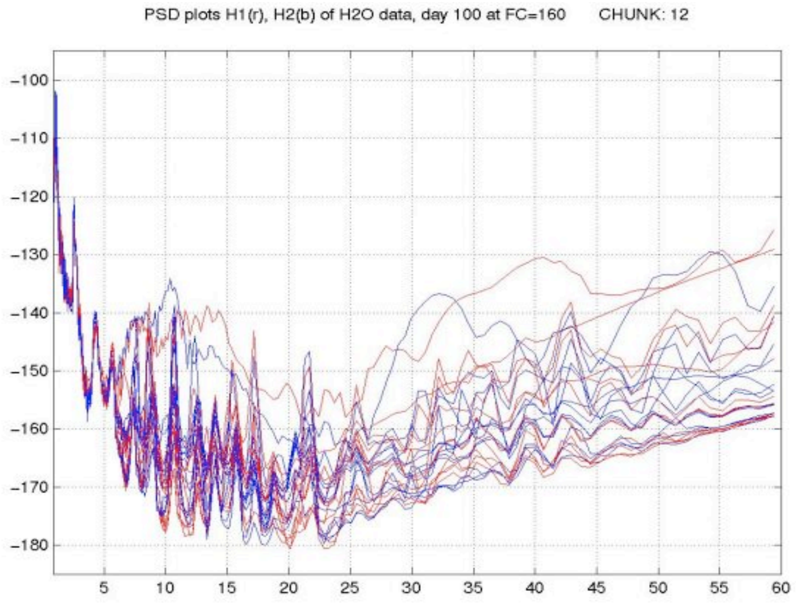


Fig. S12: linear PSD plot of vertical (Z) and horizontal (H1, H2) component spectra. **H2O** data, day 100, buried seismometer. All (12) chunks (1 chunk is approximately 2.0 hours of data).

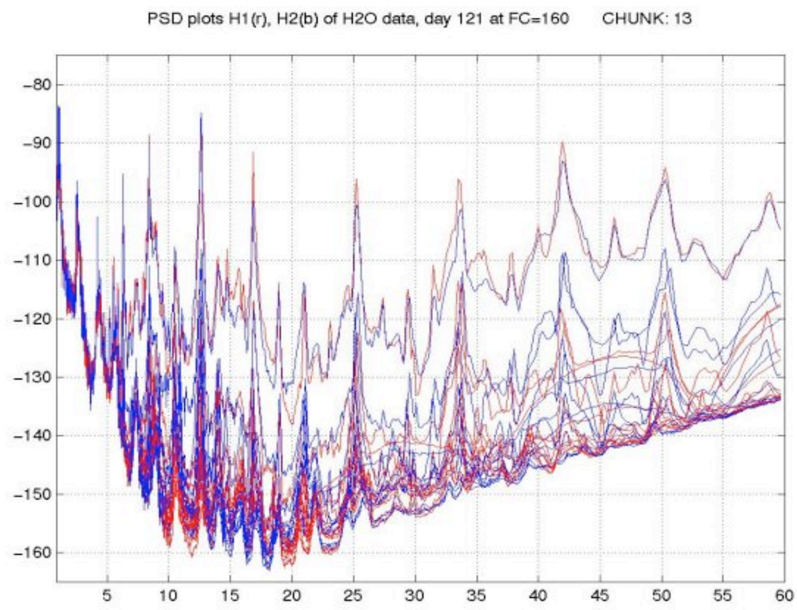


Fig. S13: linear PSD plot of vertical (Z) and horizontal (H1, H2) component spectra. **H2O** data, day 121, buried seismometer. All (13) chunks (1 chunk is approximately 1.8 hours of data).

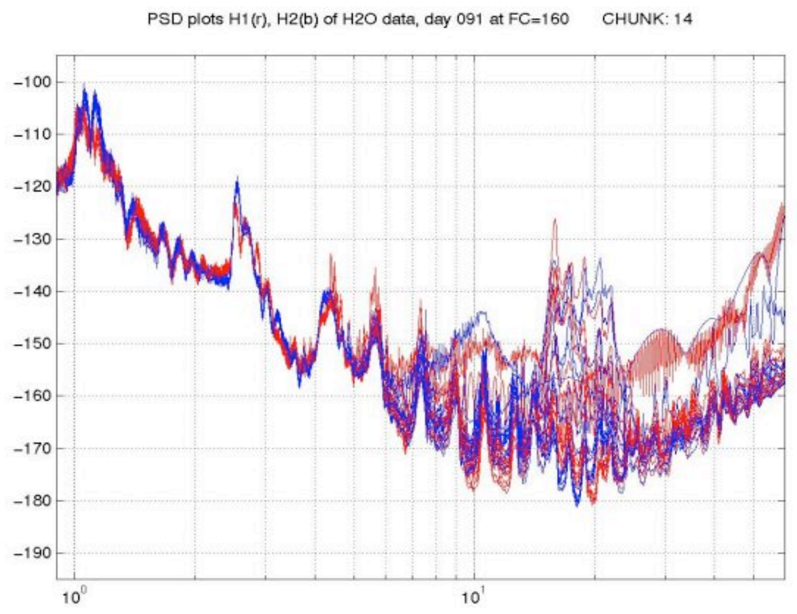


Fig. S14: logarithmic PSD plot of vertical (Z) and horizontal (H1, H2) component spectra. **H2O** data, day 091, buried seismometer. All (14) chunks (1 chunk is approximately 1.7 hours of data).

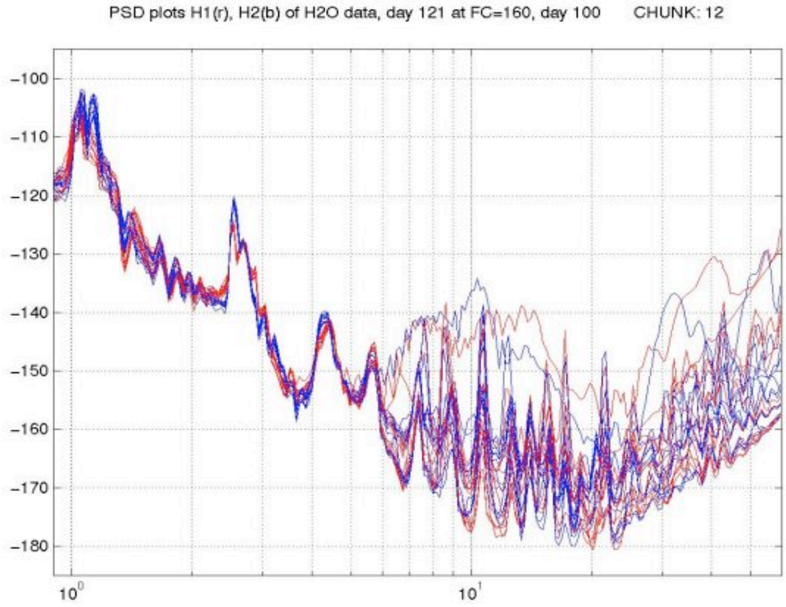


Fig. S15: logaritmic PSD plot of vertical (Z) and horizontal (H1, H2) component spectra. **H2O** data, day 100, buried seismometer. All (12) chunks (1 chunk is approximately 2.0 hours of data).

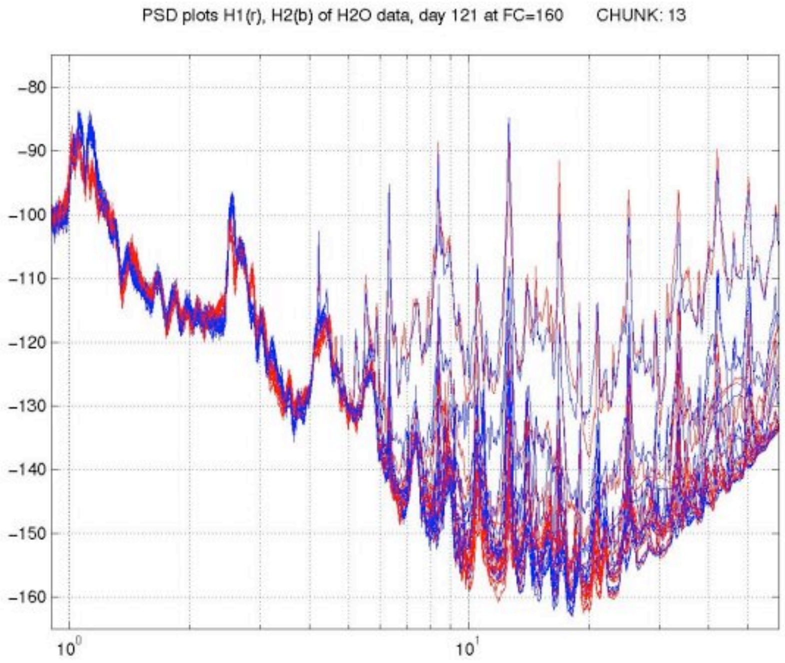


Fig. S16: linear PSD plot of vertical (Z) and horizontal (H1, H2) component spectra. **H2O** data, day 121, buried seismometer. All (13) chunks (1 chunk is approximately 1.8 hours of data).

- **`r03_polyfit5H2O***.m`**

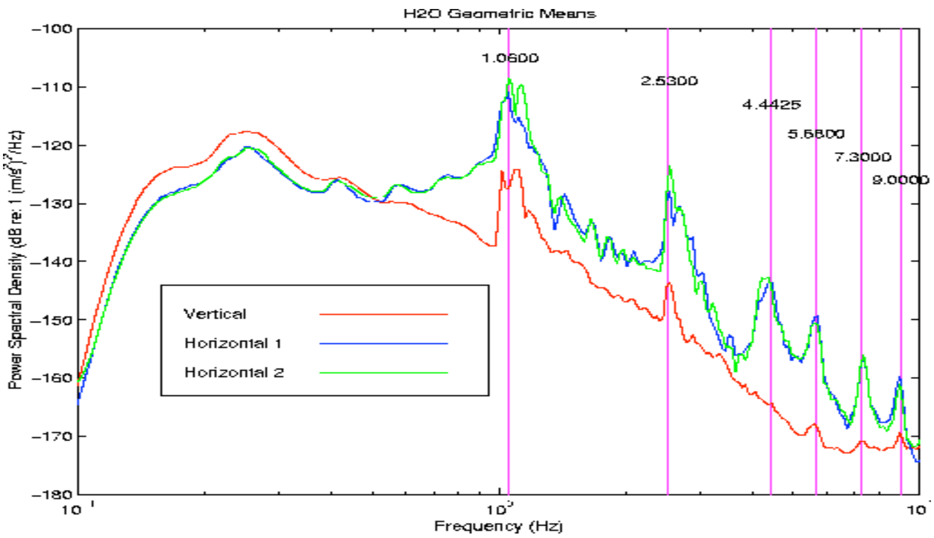


Fig. S17: The sediment resonances at **H2O** are very well established and it is possible to pick at least six unambiguous resonant frequencies (1.06, 2.53, 4.44, 5.68, 7.30, and 9.00Hz) on the horizontal component data.

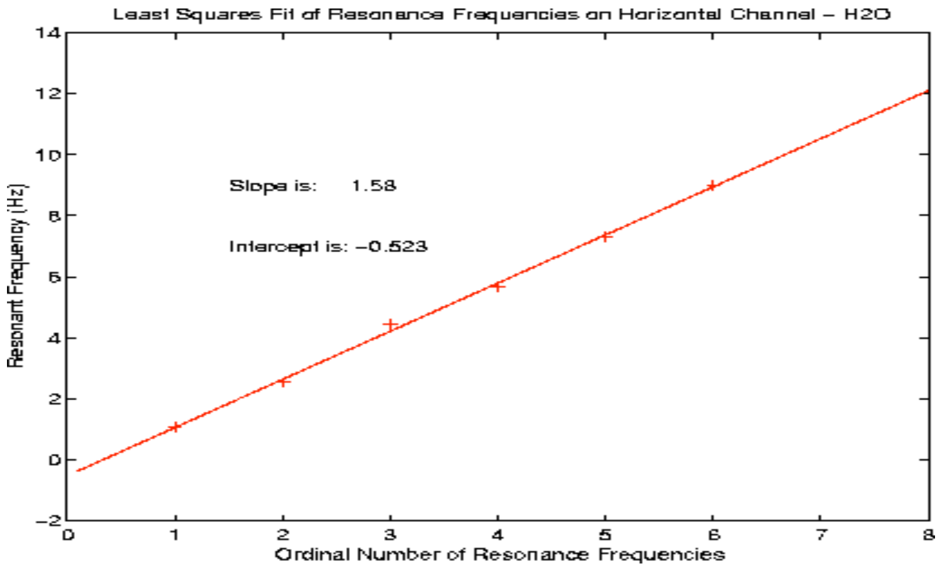


Fig. S18:A plot of the resonant frequencies from Figure S17 versus ordinal number, polyfit gives: F = 1.58; b = -0.52; ν = 0.40 and ratio = 5.28. Assuming that the uppermost shear velocity at OSN-1 is a good estimate of the uppermost shear velocity at H2O gives the sediment thickness at H2O of 13m.

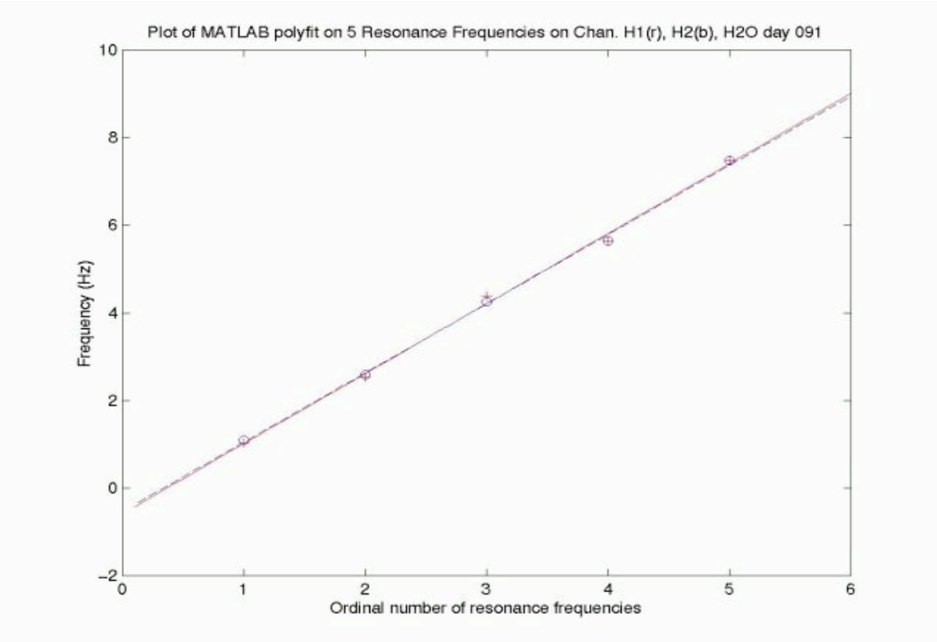


Fig. S19: A plot of 5 resonant frequencies on two horizontal channels from **H2O** day 091 versus ordinal number, polyfit gives: F = 1.57; b = -0.51; ν = 0.41 and ratio = 5.3. Again this gives a sediment thickness at H2O of 13m.

chunk	FH	bH	nuH	ratioH
H2O (day 091, picking only the first 5 resonance peaks and average of H1 & H2)				
1.	1.62	-0.61	0.33	4.83
2.	1.57	-0.51	0.40	5.2
3.	1.56	-0.49	0.44	5.7
4.	1.57	-0.52	0.39	5.1
5.	1.57	-0.51	0.40	5.2
6.	1.55	-0.48	0.45	5.7
7.	1.54	-0.46	0.47	5.8
8.	1.53	-0.42	0.42	5.2
9.	1.57	-0.51	0.40	5.2
10.	1.57	-0.51	0.40	5.2
11.	1.57	-0.52	0.39	5.1
12.	1.57	-0.52	0.39	5.1
13.	1.57	-0.50	0.42	5.2
av.	1.57	-0.51	0.41	5.32

Table S1: The parameters that are calculated to determine the sediment thickness/shear wave velocity from the picks of 5 resonance peaks in **H2O** data of day 091.

chunk	FH	bH	nuH	ratioH
H2O (day 121, picking only the first 5 resonance peaks and average of H1&H2)				
1.	1.56	-0.51	0.41	5.2
2.	1.56	-0.50	0.41	5.28
3.	1.57	-0.50	0.42	5.2
4.	1.56	-0.50	0.42	5.3
5.	1.57	-0.51	0.40	5.2
6.	1.57	-0.53	0.38	5.1
7.	1.58	-0.55	0.37	5.02
8.	1.55	-0.51	0.40	5.2
9.	1.57	-0.53	0.37	5.1
10.	1.56	-0.48	0.45	5.7
11.	1.55	-0.47	0.46	5.74
12.	1.56	-0.49	0.44	5.7
13.	1.56	-0.49	0.44	5.7
av.	1.56	-0.51	0.41	5.28

Table S2: The parameters that are calculated to determine the sediment thickness/shear wave velocity from the picks of 5 resonance peaks in **H2O** data of day 121.

- **r04_polyfit13H2O***.m**

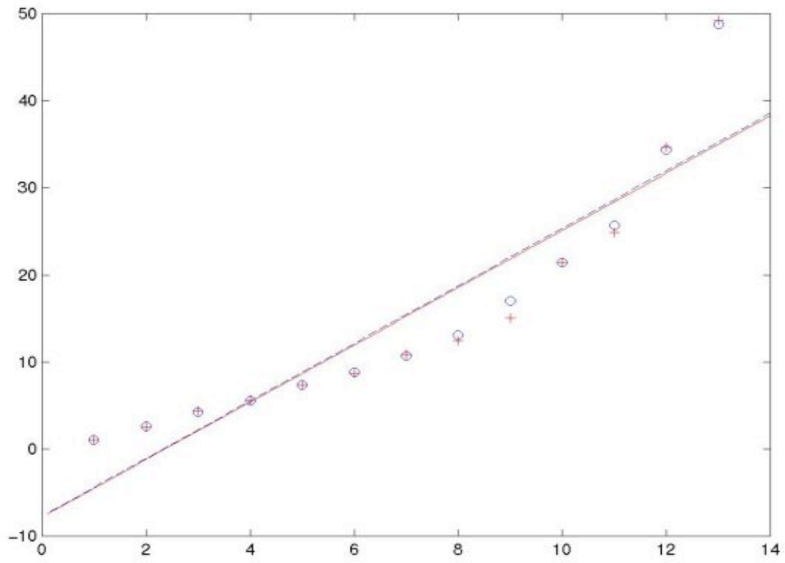


Fig. S20: A plot of 13 (resonant) frequencies from **H2O** day 121 versus ordinal number. Chunk 1.

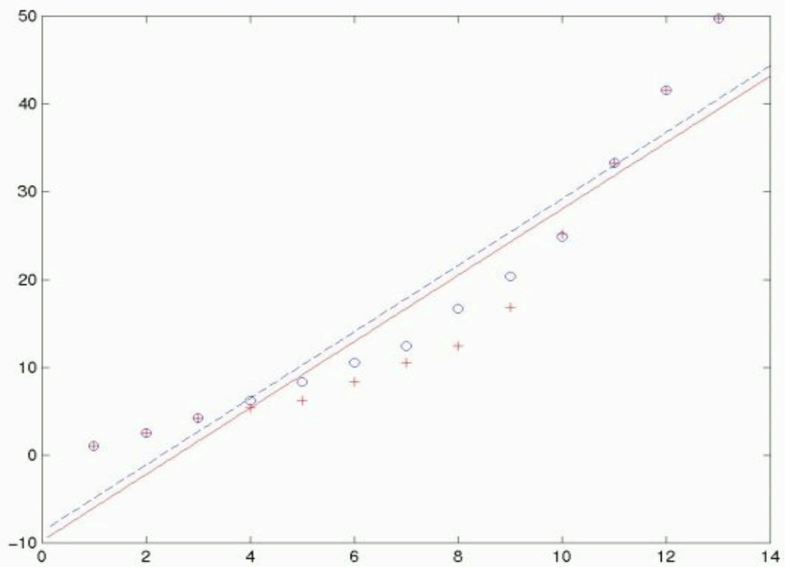


Fig. S21: A plot of 13 (resonant) frequencies from **H2O** day 121 versus ordinal number. Chunk 9.

chunk	FH1	FH2	bH1	bH2	nuH1	nuH2	ratioH1	ratioH2
H2O (day 121, picking 13 resonance peaks from H1 & H2 plots)								
1	1.61	1.60	-0.60	-0.61				
2	1.59	1.59	-0.59	-0.60				
3	1.59	1.58	-0.57	-0.55				
4	1.60	1.58	-0.59	-0.50				
5	1.55	1.57	-0.48	-0.54				
6	1.56	1.54	-0.48	-0.47				
7	1.54	1.55	-0.45	-0.50				
8	1.57	1.57	-0.50	-0.54				
9s	1.50	1.51	-0.52	-0.55				
10	1.56	1.55	-0.46	-0.49				
11	1.54	1.56	-0.46	-0.50				
12	1.57	1.55	-0.50	-0.47				
13	1.57	1.56	-0.50	-0.53				
av.	1.57	1.57	-0.52	-0.52	0.40	0.40	5.23	5.23

Table S3: The parameters that are calculated to determine the sediment thickness/shear wave velocity from the picks of 13 resonance peaks in **H2O** data of day 121.

For Co = 24.5m/s this would infer a sediment thickness of H = 13m
For Co = 30m/s this would infer a sediment thickness of H = 18m
For Co = 40m/s this would infer a sediment thickness of H = 29m
For Co = 50m/s this would infer a sediment thickness of H = 42m
For Co = 60m/s this would infer a sediment thickness of H = 57m
For Co = 70m/s this would infer a sediment thickness of H = 74m
For Co = 80m/s this would infer a sediment thickness of H = 92m
For Co = 90m/s this would infer a sediment thickness of H = 112m

Models giving dispersion curves that closely match those from ocean-bottom observations are characterised by a seabed S-wave velocity of 30-40m/s and a velocity gradient between 10 and 20m/s in the top few metres of the sediment column [*Whitmarsh and Lilwall,* 1982]. Therefore it is highly unlikely that the seabed S-wave velocity exceeds 90m/s at the H2O site. We can conclude that according to the calculations made on the basis of the used paper, the sediment thickness is max.18m.

- **r05_polyfitOSN.m**

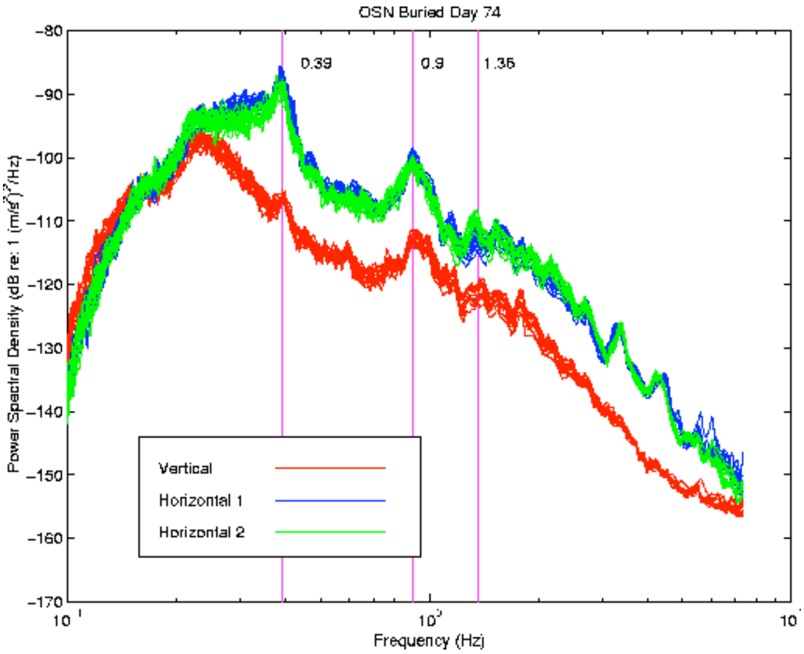


Fig. S22: The resonant frequencies for **OSN** are 0.39, 0.90, and 1.36Hz as shown in this figure of the spectra for the buried seismometer at OSN-1.

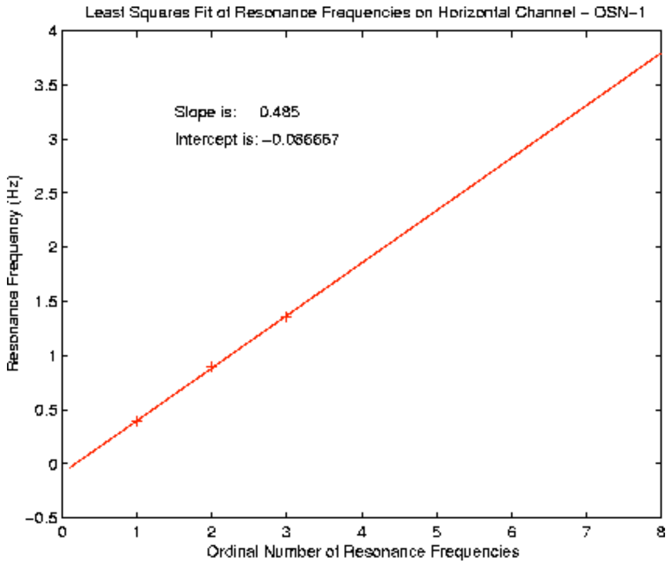


Fig. S23: A plot of the resonant frequencies from Figure S21 versus ordinal number gives: F = 0.485, b = -0.087, ν = 0.562, ratio =2.22. Using the known sediment thickness at OSN-1 of 242m we can estimate the uppermost shear velocity as 24.5m/s.

OSN day

Resonance peaks for the Shallow Buried seismometer.

Sh1 0.39
Sh2 0.90
Sh3 1.36

chunk	FH	bH	nuH	ratioH
OSN (day 74)				
24-hour	0.49	-0.087	0.562	2.22
With a sediment thickness of 242m this infers a Co of 24.5m/s				

Table S4: The parameters that are calculated to determine the sediment thickness/shear wave velocity from the picks of 3 resonance peaks in **OSN** data of day 74.

OSN day 74 + OSN day 126

Resonance peaks for the Shallow Buried seismometer.

Sh1 0.39
Sh2 0.90
Sh3 1.21 same
Sh4 1.70
Sh5 2.21

The Borehole seismometer is approx. the same at the first three peaks, but the 3th and 4th are not clearly visible. The Seafloor seismometer is not useful for determining resonance peaks. There are 3 peaks but at lower frequencies. In this figure the vertical noise is higher than horizontal noise. Other noise sources which affect the signal might be stronger here and could distort the resonance peaks.

chunk	FH	bH	nuH	ratioH
OSN (day 74 & 126, from sheets)				
24-hour	0.44	-0.050	0.66	2.44
With a sediment thickness of 242m this infers a Co of 17.6m/s				

Table S5: The parameters that are calculated to determine the sediment thickness/shear wave velocity from the picks of 5 resonance peaks in **OSN** data of day 74 and 126.

- **`r06_pickallrfreqH2O***.m`**
- **`r07_plotallrfreqH2O***.m`**

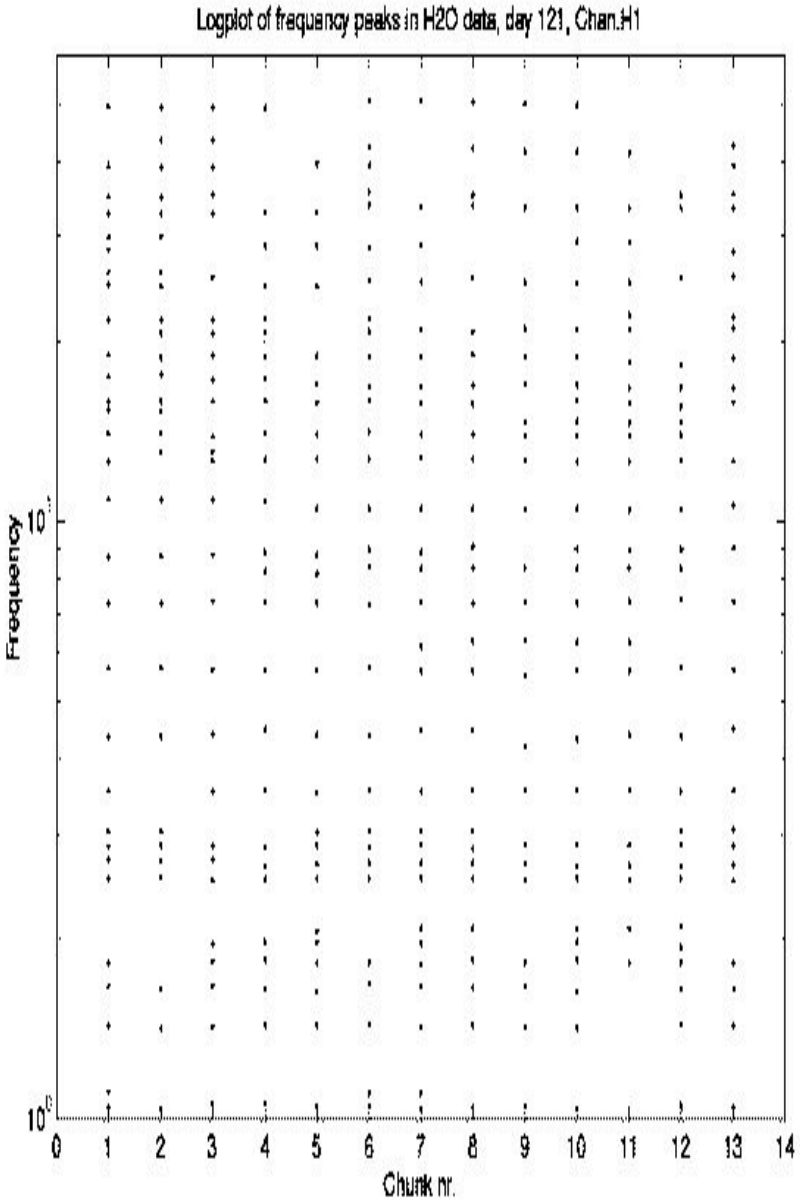


Fig. S24: A plot of 30 picked (resonance) frequency peaks for each chunk (total of 13) of data from day 121 of the **H2O** site.

- **r08_plotchunkrfreqH2O.m**

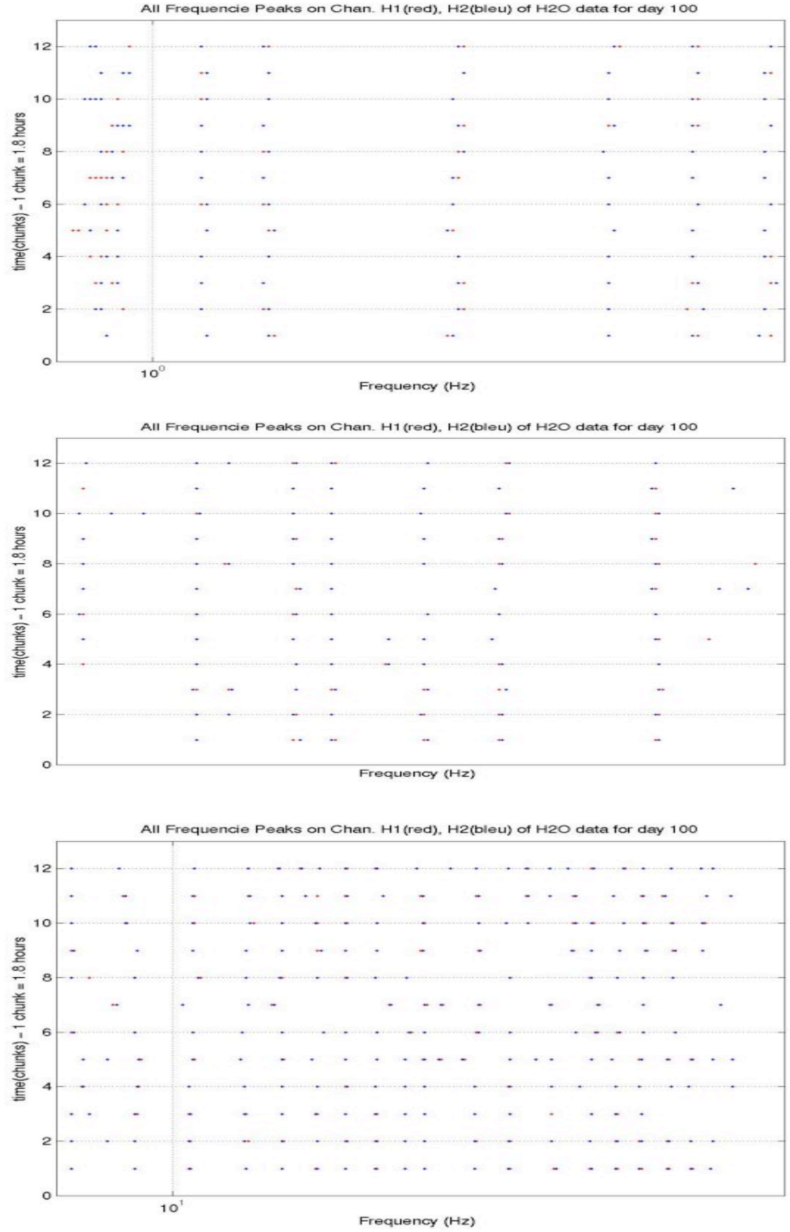


Fig. S25: A logaritmic plot (in three parts) of plots 30 picked (resonance) frequency peaks for each chunk (total of 13) of data from day 100 of the **H2O** site.

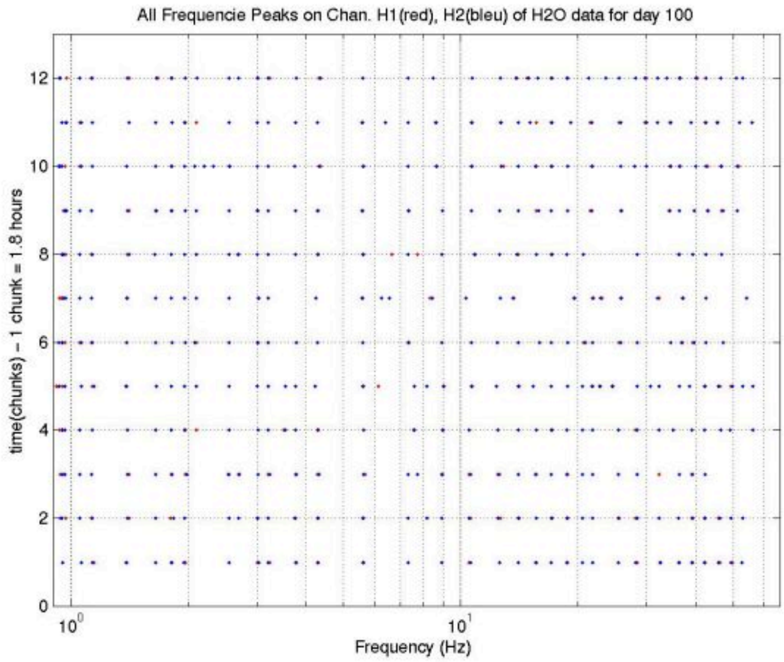
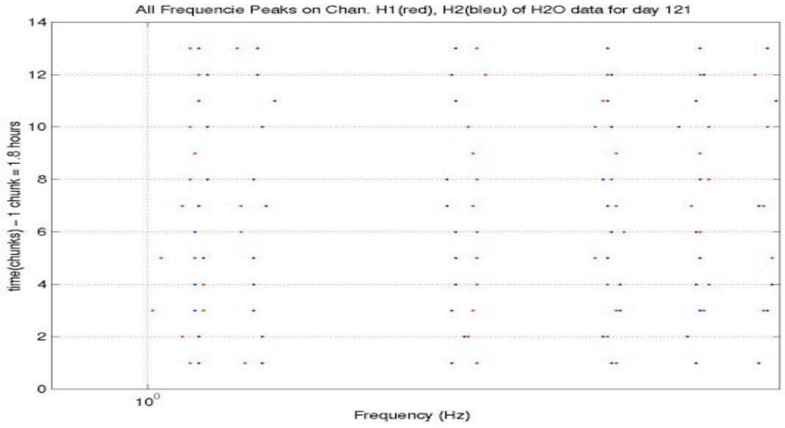
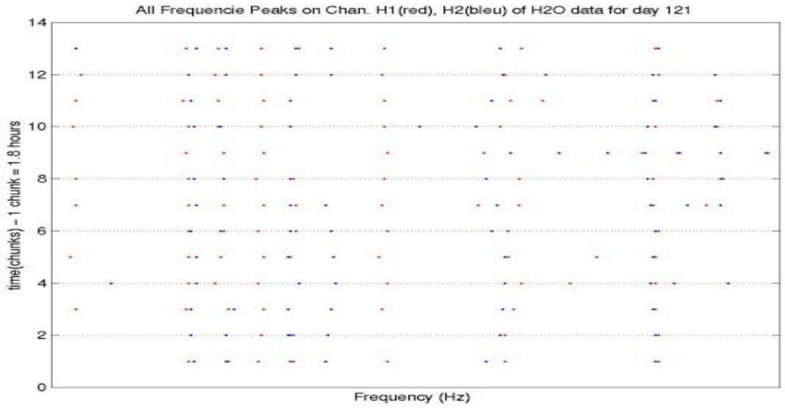


Fig. S26: A logaritmic plot of plots 30 picked (resonance) frequency peaks for each chunk (total of 13) of data from day 100 of the **H2O** site.


All Frequencie Peaks on Chan. H1(red), H2(bleu) of H2O data for day 121
time(chunks) - 1 chunk = 1.8 hours
Frequency (Hz)


All Frequencie Peaks on Chan. H1(red), H2(bleu) of H2O data for day 121
time(chunks) - 1 chunk = 1.8 hours
Frequency (Hz)

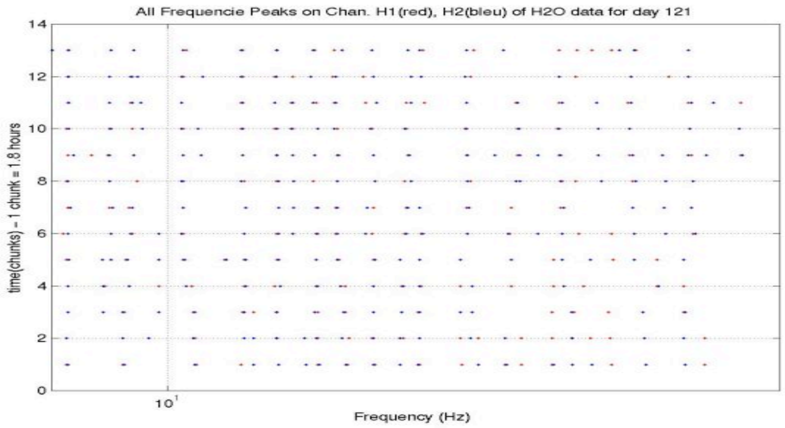

All Frequencie Peaks on Chan. H1(red), H2(bleu) of H2O data for day 121
time(chunks) - 1 chunk = 1.8 hours
Frequency (Hz)

Fig. S27: A logaritmic plot (in three parts) of plots 30 picked (resonance) frequency peaks for each chunk (total of 14) of data from day 121 of the **H2O** site.

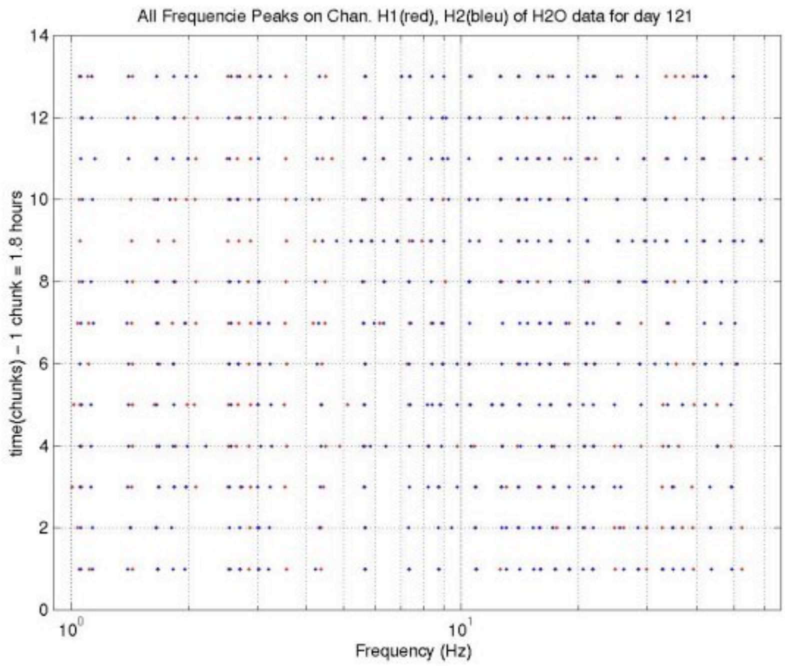
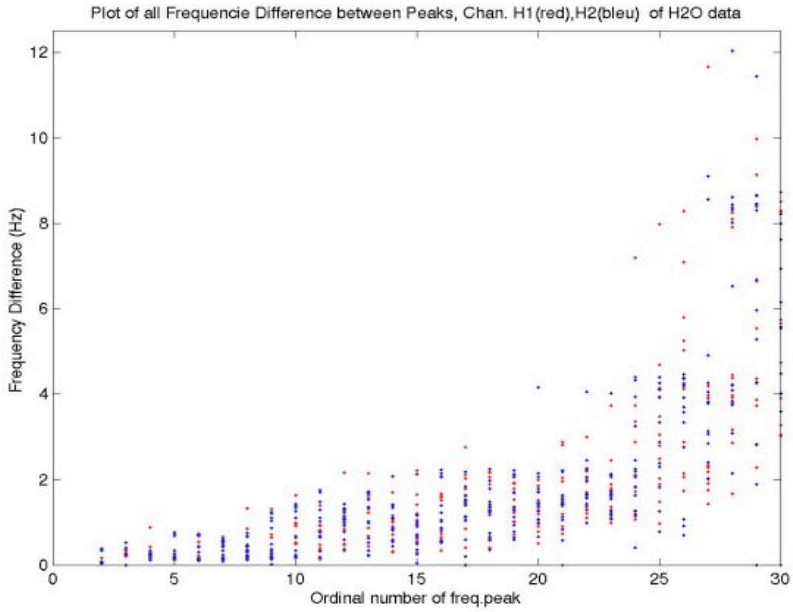


Fig. S28: A logaritmic plot of plots 30 picked (resonance) frequency peaks for each chunk (total of 14) of data from day 121 of the **H2O** site.

- **`r09_plot29rfreqdifH2O***.m`**
- **`r10_trytick.m`**



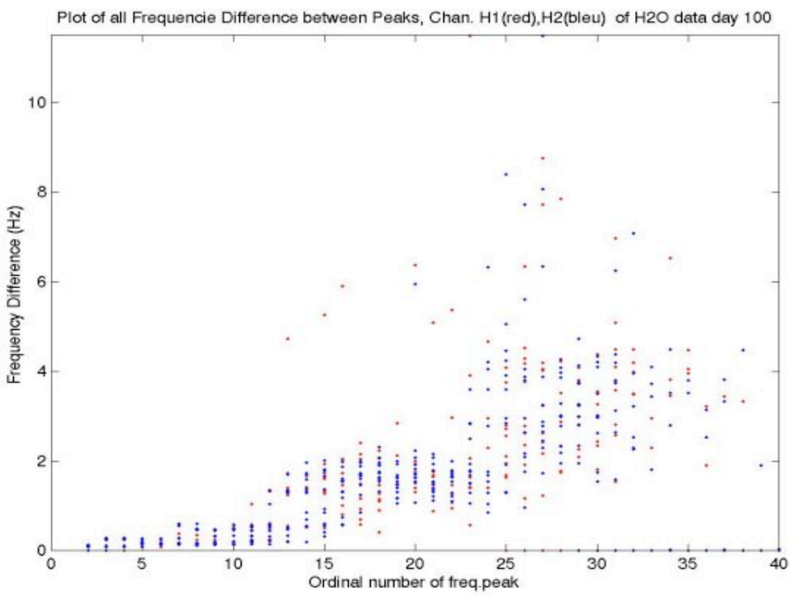


Fig. S29: plots the differences between 29 (top figure) and 39 (bottom figure) (resonance) frequency peaks for all chunks of data from day 121 / 100 of the **H2O** site. The picked peaks were not always adjacent, that's why sometimes very high differences occur. The M-file is not ideal for this purpose.

- **r11_colorarrayH2O***.m**
- **r12_colorplotH2O***.m**

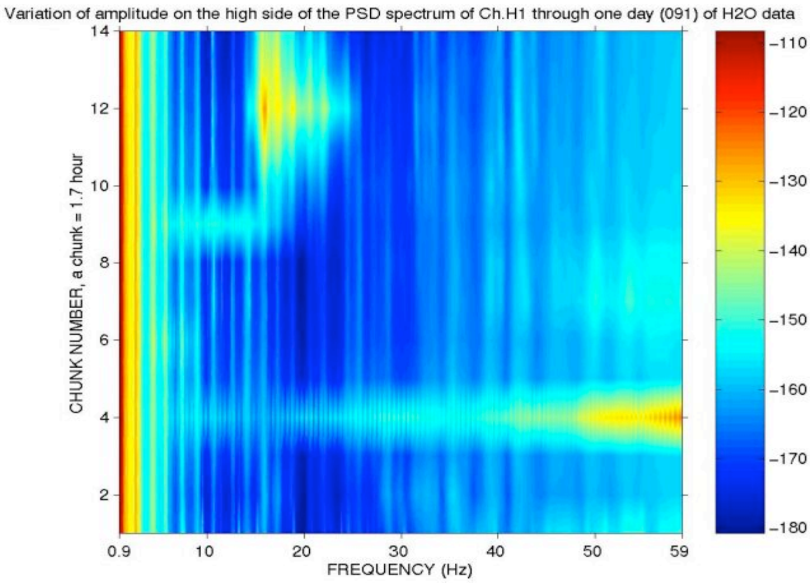


Fig. S30: Linear plot of the variation of PSD amplitude of channel H1 throughout one day (091) of data.

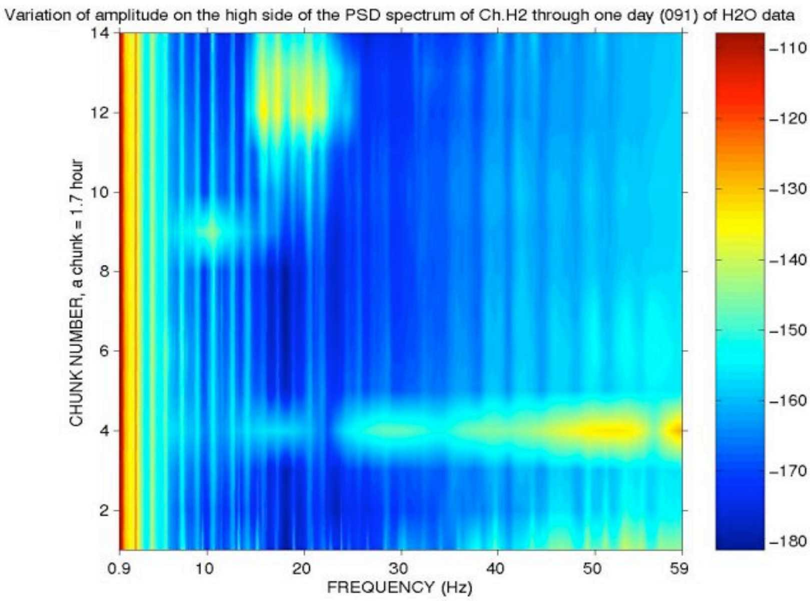


Fig. S31: Linear plot of the variation of PSD amplitude of channel H2 throughout one day (091) of data.

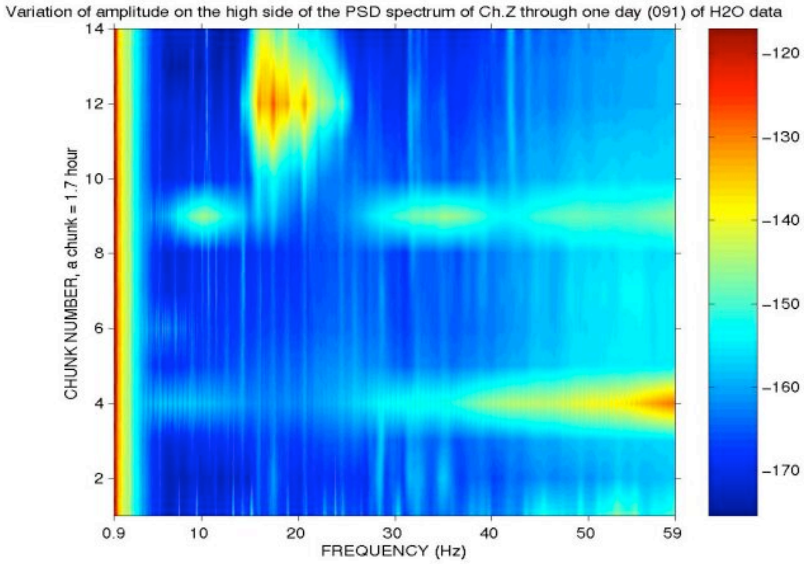


Fig. S32: Linear plot of the variation of PSD amplitude of channel Z throughout one day (091) of data.

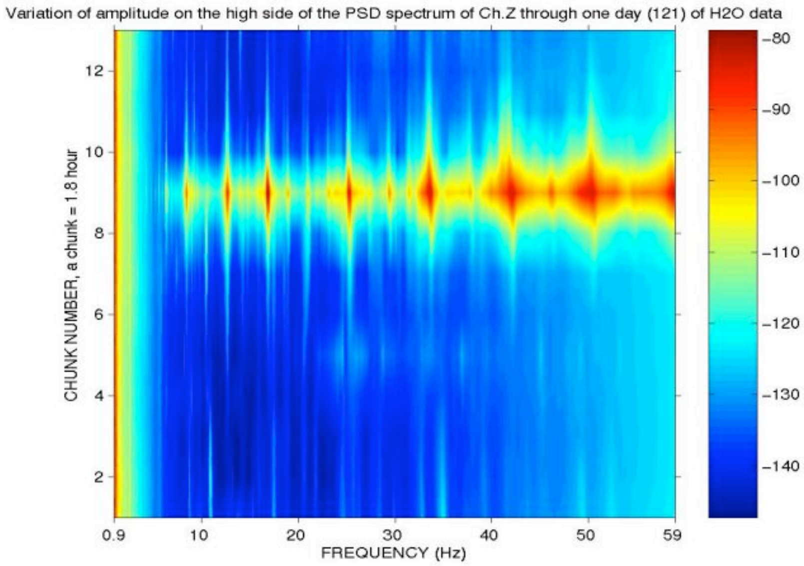


Fig. S33: Linear plot of the variation of PSD amplitude of channel Z throughout one day (121) of data.

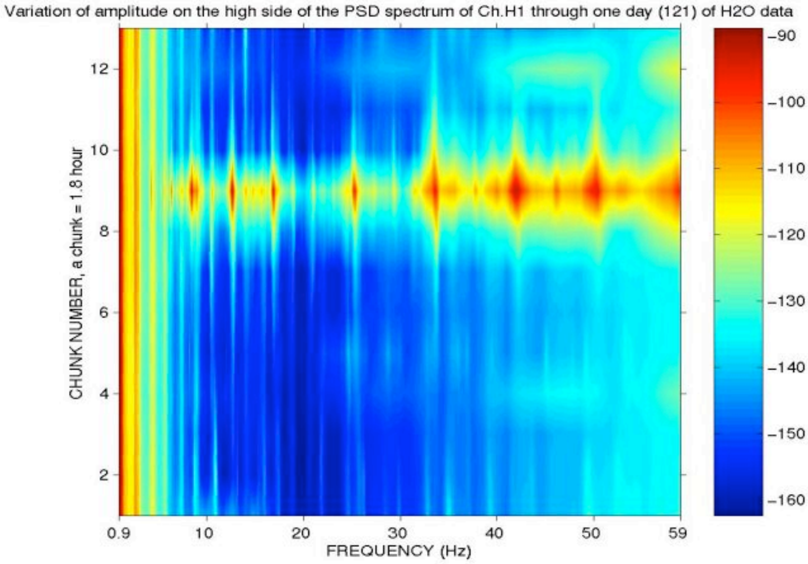


Fig. S34: Linear plot of the variation of PSD amplitude of channel H1 throughout one day (121) of data.

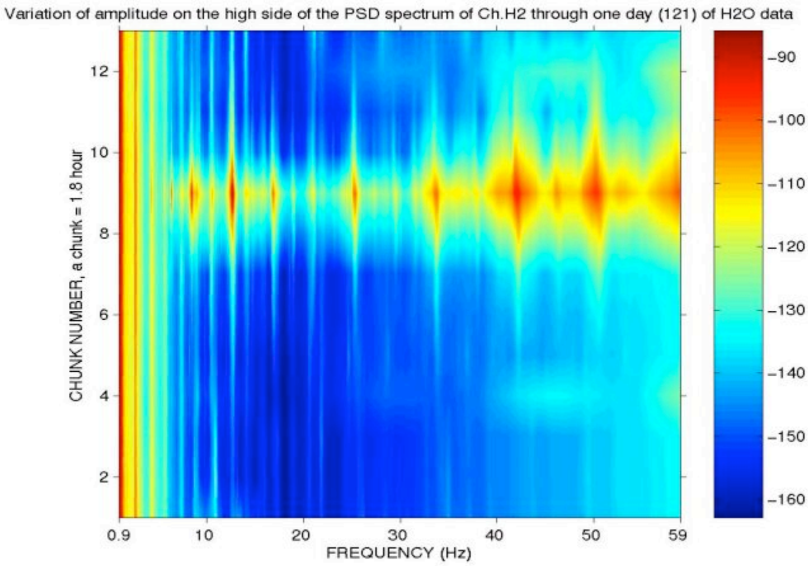


Fig. S35: Linear plot of the variation of PSD amplitude of channel H2 throughout one day (121) of data.

- **r13_logcolorplotH2O***.m**

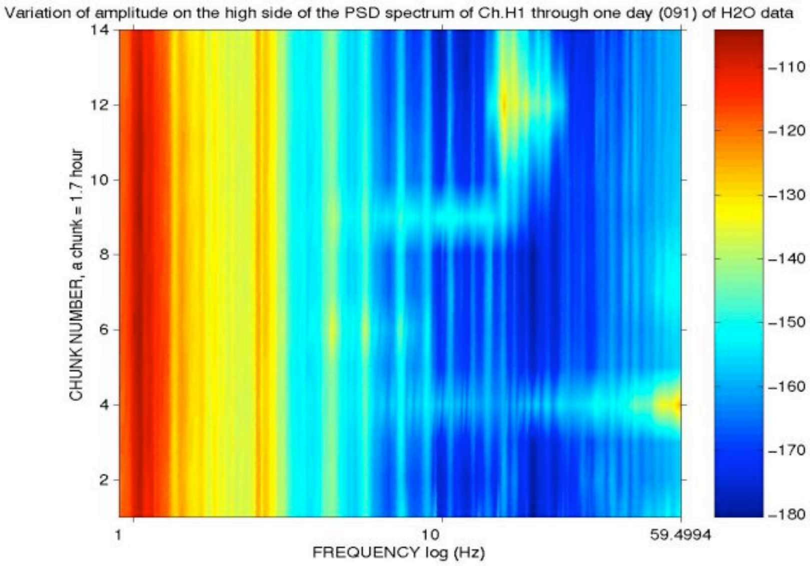


Fig. S36: Logarithmic plot of the variation of PSD amplitude of channel H1 throughout one day (091) of data.

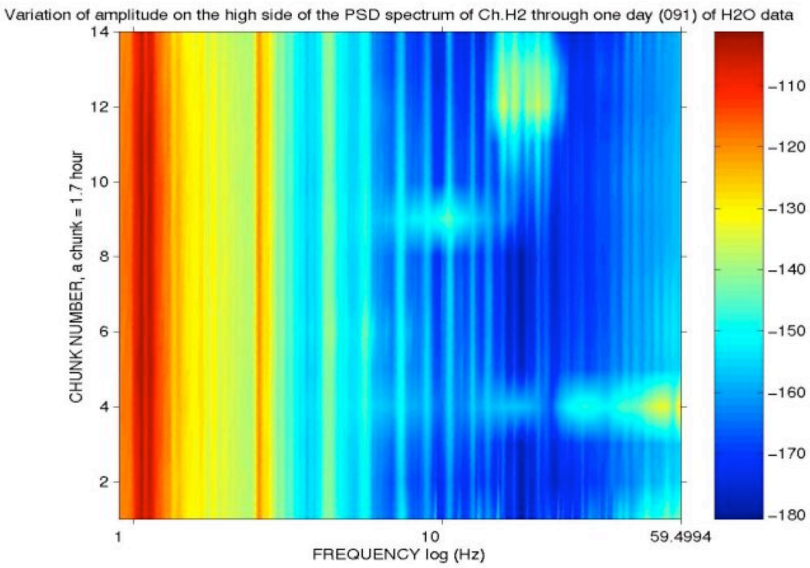


Fig. S37: Logarithmic plot of the variation of PSD amplitude of channel H2 throughout one day (091) of data.

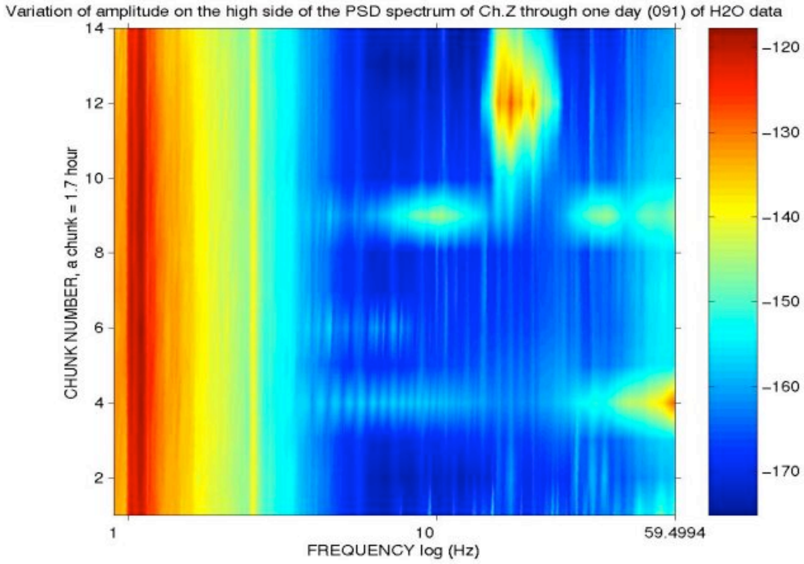


Fig. S38: Logarithmic plot of the variation of PSD amplitude of channel Z throughout one day (091) of data.

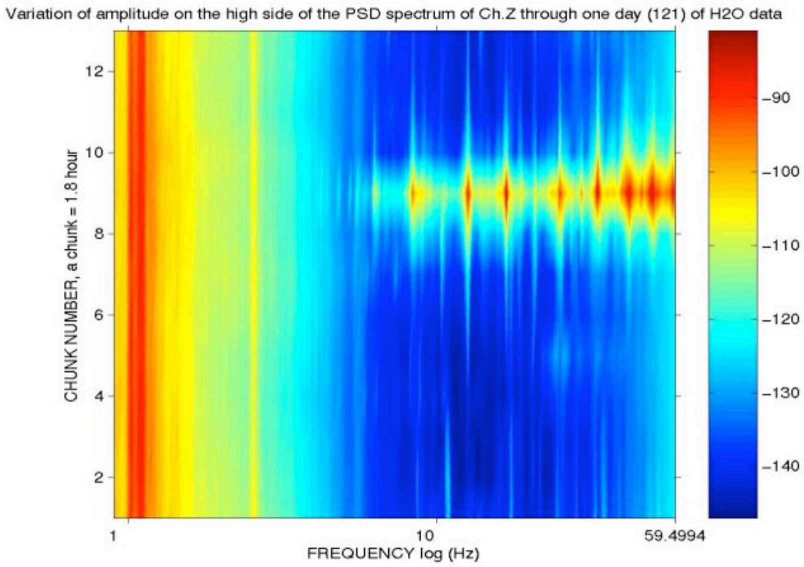


Fig. S39: Logarithmic plot of the variation of PSD amplitude of channel Z throughout one day (121) of data.

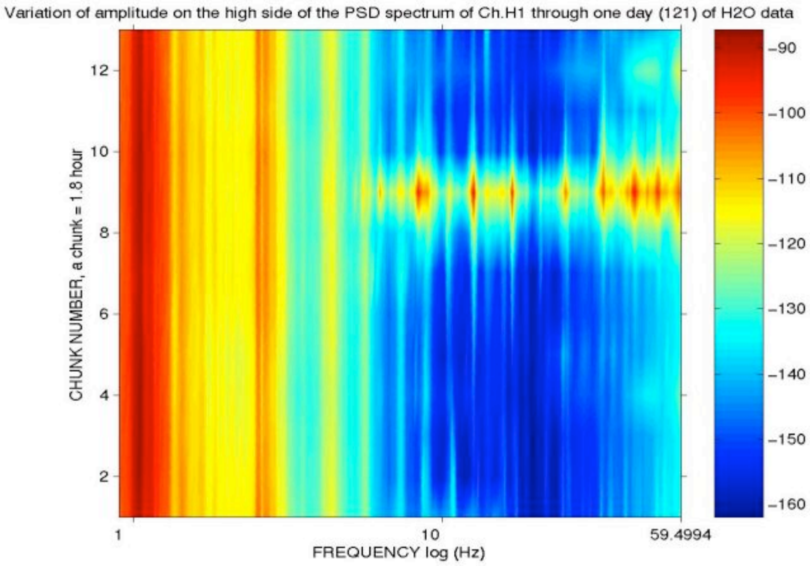


Fig. S40: Logarithmic plot of the variation of PSD amplitude of channel H1 throughout one day (121) of data.

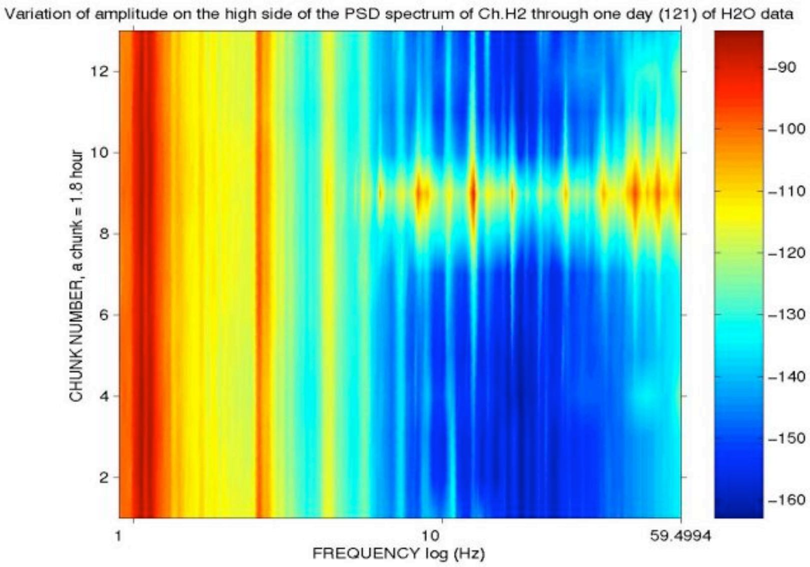


Fig. S41: Logarithmic plot of the variation of PSD amplitude of channel H2 throughout one day (121) of data.

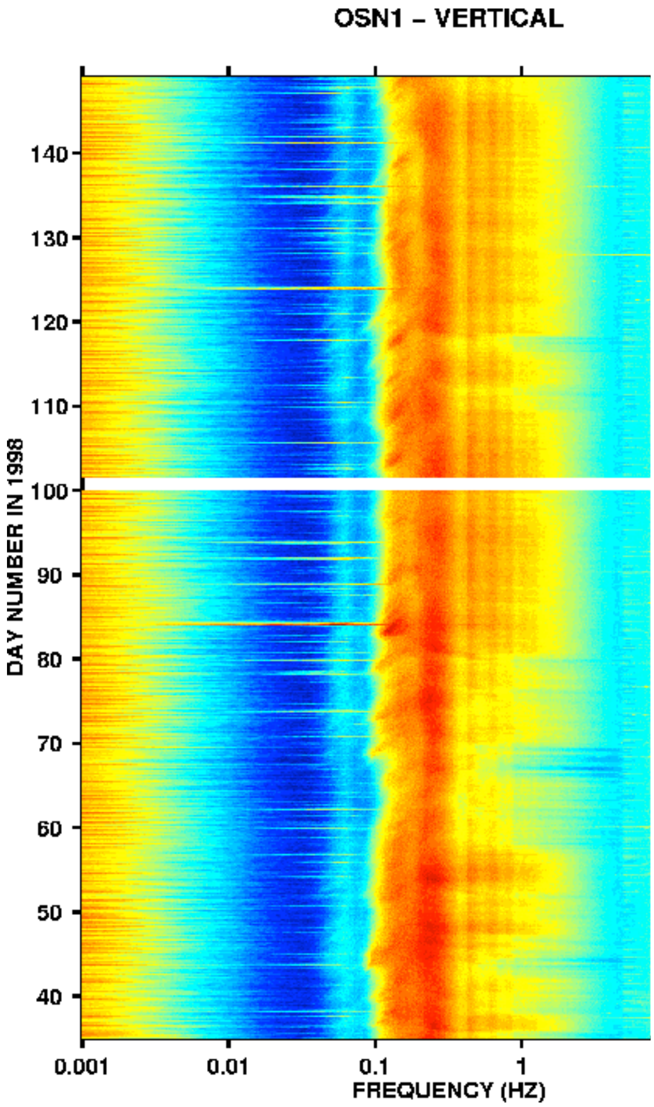


Fig. S42: This true amplitude spectrogram summarizes the vertical component ambient noise in the band 1mHz to 7.5Hz for over 110days of the broadband borehole seismometer deployment at **OSN-1**.

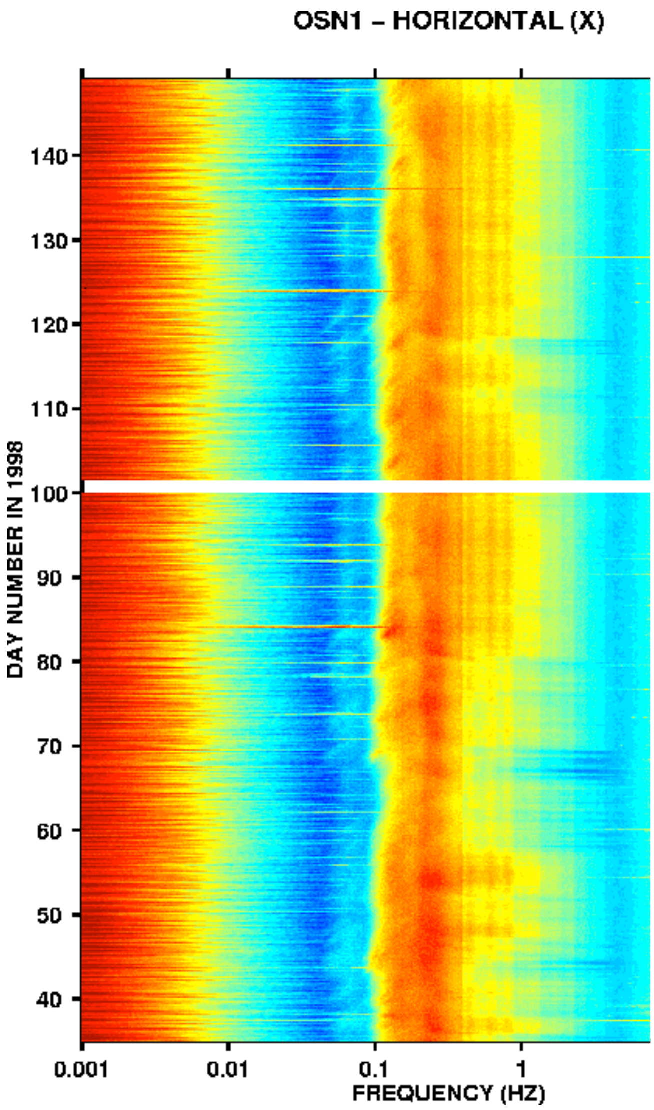


Fig. S43: This true amplitude spectrogram summarizes the horizontal component ambient noise in the band 1mHz to 7.5Hz for over 110days of the broadband borehole seismometer deployment at **OSN-1**.

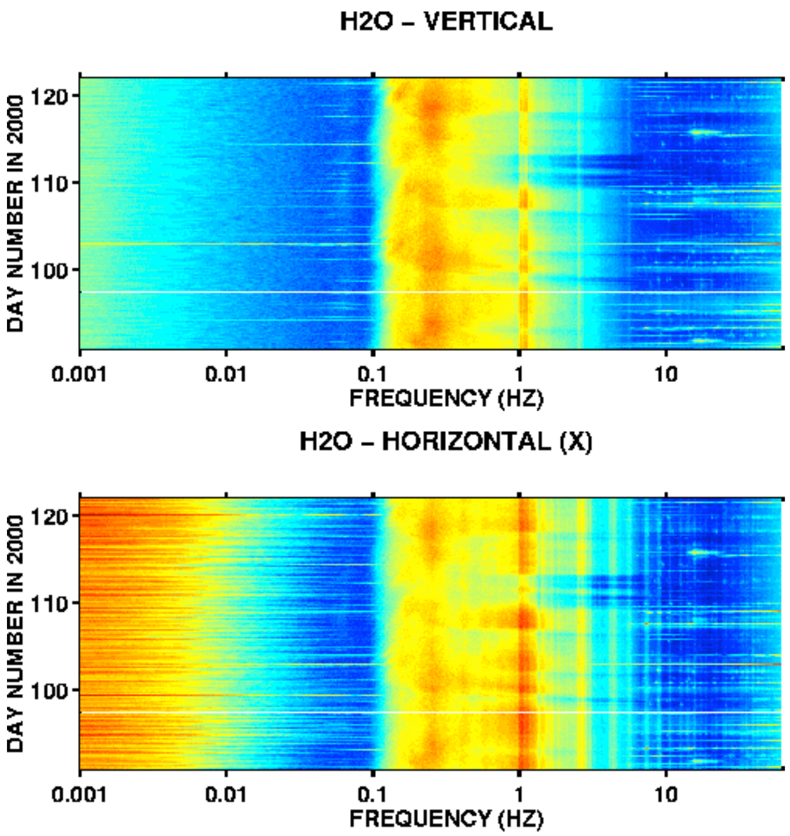


Fig. S44: vertical and horizontal component spectrograms are shown for 1 month of data from the new **H2O** site. These figures have the same time, frequency and color scales as Figures S42 and S43 for OSN-1. The sediment resonances are more clearly defined and extend to higher frequencies on the horizontal component data.

Discussion and Conclusions

18. What does the short-period psd spectrum look like?
(Plot the short-period spectrum for several days of H2O data, Figs. S8 t/m S16)

- r01_FC160psdH2O***.m
- r02_FC160tomH2O***.m

The **H2O** buried seismometer short period spectrum shows:

✓ **The vertical channel is generally 10-20 dB quieter than the horizontals.**

✓ **The spectrum shows big, approximately equally spaced peaks in the microseism band.**

✓ **The spectrum shows smaller, more variable peaks in the VLF band. But the highest peaks still seem to be roughly equally spaced. Through one day the psd of a channel may vary by 50 dB at one frequency in this band. This is probably caused by shipping (day121).**

✓ **For days 091 and 100 the noise floor for al three components is about -180 dB and is reached in the VLF band (around 11 Hz). For day 121 the noise floor is lower for the horizontal channels (but located at the same place), it comes down to -160 dB.**

✓ **For days 091 and 100 the noise roof for the horizontal channels is located at the first resonance peak (1.0Hz), at -100 dB. The vertical channels have two approximately equal peaks, one at around 0.2 Hz (microseism peak) and one at the first resonance peak.**
For day 121 the noise roof for the horizontal channels is the first resonance peak (-85 dB),
for the vertical channels it's the microseism peak at -80 dB.

✓ **Day 091 has one chunk of very noisy incoherent signals on all channels (Figure S11)**

19. Where are the resonance peaks?
(Pick all peaks in the PSD plots of OSN and H2O data, Figs. S17 t/m S23)

- r03_polyfit5H2O***.m
- r04_polyfit13H2O121.m
- r05_polyfitOSN.m

✓ There are at least 6 clearly visible resonance peaks in the **H2O** data, which are located at 1.06, 2.53, 4.44, 5.68, 7.30 and 9.0Hz on the horizontal component spectra (Figure S17). The first two peaks are also visible on the vertical component spectra.

✓ There are at least 3 clearly visible resonance peaks in the **OSN** data, which are located at 0.39, 0.90 and 1.36Hz on the horizontal component spectra (Figure S22). The first two peaks are also visible on the vertical component spectra.

✓ It is interesting to see that there might be more important resonance peaks than the 5 mentioned in the paper of *Godin & Chapman*. The new and real-time H2O data provide a great amount of data to study these phenomena in more detail.

20. What does this infer for the sediment thickness?

✓ We used a polyfit command to calculate the best-fit line to the f_n, with a function f_n = Fn - b. Then the method described in the paper of Godin and Chapman (previous chapter) is used to determine the thickness of the sediments. Using various days and picking 5 to 13 peaks we always arrive at a sediment thickness of approximately 13m. (Tables S1 t/m S5)

21. Are the resonance peaks constant throughout one day of data?
(Plot the data of the H2O site in a colour spectrogram to determine how the (resonance) peaks vary over 24-hours)

- r06_pickallrfreqH2O121.m
- r07_plotallrfreqH2O121.m
- r08_plotchunkrfreqH2O121.m
- r09_plot29rfreqdifH2O121.m
- r11_colorarrayH2O***.m
- r12_colorplotH2O***.m
- r13_logcolorplotH2O***.m

Looking at Figures S30 t/m S35 and S36 t/m S41:

✓ Looking at the linear plots of the variation of the PSD throughout day 091 of **H2O** data, at least 11 peaks are visible in the horizontal spectrum. They are spaced approximately equally and are constant throughout the day. They are located in the band of 0.9-16Hz. At two moments during the day (chunks 4 and 12) peaks came up at 20Hz and around 50Hz, but these only lasted for a few hours. They are probably the result of man-made noise or biological noise (whales or fish). The vertical spectrum doesn't show all the resonance peaks but enhances the noise at 20Hz and around 50Hz. Also it shows that the 8th chunk of vertical data was very incoherent and noisy.

✓ Looking at the linear plots of the variation of the PSD throughout day 121 of **H2O** data, at least 8 peaks are visible in the horizontal spectrum. They are spaced approximately equally and are constant throughout the day. They are located in the band of 0.9-12Hz. At one moment during the day (chunk 7 t/m 11) peaks comes up throughout a big part of the VLF band (5-59Hz), this is probably caused by the passing of a ship. The vertical spectrum doesn't show all the resonance peaks but does show the peaks in the VLF band.

✓ Looking at the logarithmic plots of the variation of the PSD throughout day 091 of **H2O** data, at least 11 peaks are visible in the horizontal spectrum. They are constant throughout the day, and show nicely the steep sides of the first 3 peaks. The vertical spectrum doesn't show all the resonance peaks but does show the peaks in the VLF band.

✓ Looking at the logarithmic plots of the variation of the PSD throughout day 121 of **H2O** data, at least 8 peaks are visible in the horizontal spectrum. They are constant throughout the day, and show nicely the steep sides of the first 3 peaks. The vertical spectrum doesn't show all the resonance peaks but does show the peaks in the VLF band.

22. Are the resonance peaks constant throughout several days of data?
(Plot several days of H2O data in colour spectrograms to determine how the (resonance) peaks vary with different days)

Looking at Figures S42 t/m S44:

- ✓ The true amplitude spectrograms of **OSN -1** are a summary of the ambient noise in the frequency band of 0.001-7.5Hz over 110 days of data. At least 4 peaks are visible throughout the period, between 0.1-1Hz. This doesn't correspond to the peaks found in Figure 22. In Figure 22 there were only two peaks between 0.1-1Hz, and they were located at 0.39Hz and 0.9Hz and also a third one at 1.36Hz. The horizontal and vertical spectrograms are very similar, but the horizontal spectrogram has much higher noise in the infragravity band below 0.01Hz (caused by tilting of the instrument).
- ✓ The true amplitude spectrogram of **H2O** is a summary of the ambient noise in the frequency band of 0.001-70Hz over 1 month of data. At least 10 peaks are visible throughout the period on the horizontal spectra, between 1.0 and 20Hz. This corresponds to the peaks found in Figures S17 and S30 t/m S35. The vertical spectrogram only shows the first three peaks and has much lower noise levels in the infragravity band (less influenced by tilt noise).

What is the thickness of the sediments at the H2O site???

For shallow buried and seafloor sensors, shear wave resonances in the sediments are the dominant mechanism contributing to the ambient noise in the band 0.3-10Hz. These peaks caused by shear wave resonances in the sediment layer, are clearly visible in all the horizontal spectra I have looked at.

Using the theory of *Godin & Chapman* a sediment thickness of 13m is calculated for the H2O site. This is might be the thickness to the first interbedded chert layer. This kind of conclusion would agree with the seismic profiles taken in the area of the H2O site (Figs. A2 t/m A14). However there were some important assumptions made before we arrived at this thickness. It is assumed that the shear wave velocity at the H2O site is the same as at the OSN site. This might not be true. Furthermore the shear velocity that we have assumed for the OSN site is also not certain. It is again calculated from the (three) shear wave resonance peaks at this site.

To know either the shear wave resonance at the H2O or OSN site would help better constrain the sediment thickness. If the proposed drilling at the H2O site will take place, it will hopefully give some clear-cut geological evidence of the applicability of shear wave resonance peaks to determine sediment thickness.

Appendix A
The amplitude of body wave phases at teleseismic distances

There are several magnitude scales used to describe the size of earthquakes:

- m_b describes the amplitude of the short-period body waves near 1 Hz
- M_s is derived from the amplitude of 20-s surface waves. Earthquakes as large as $M_s = 6.5$ are relatively common (about 50 per year), and earthquakes larger than $M_s = 7.5$ occur only a few times per year [*Bolt,* 1976]. The number of earthquakes per unit time falls roughly by a factor of ten for each unit of magnitude [*Scholtz*, 1990].
- M_w is related to the earthquake moment

The relationship of these scales to more fundamental measures of earthquake size is complex and problematical. The amplitude of a teleseismic body wave phase depends on the details of the earthquake source, the instrument location and ray path, and the geology at the receiver. The practical definition of a "detection limit" is *the smallest magnitude at a particular range* for which it is likely that an earthquake phase will be detected with sufficient signal to noise to be accurately picked. The models demonstrate some key differences between continental and oceanic sites:

- The detection limit for teleseismic, short-period body waves is much higher at the typical continental sites
- The detection limit for local events is often lower at oceanic sites
- The long-period body wave detection limit is similar at continental and ocean floor sites

The difference in detection limits is due to the noise climate rather than to different propagation characteristics under the ocean.

Appendix B
Laws of physics applicable to this subject

Hooke's Law
Law of elasticity discovered by the English scientist Robert Hooke in 1660, which states that, for relatively small deformations of an object, the displacement or size of the deformation is directly proportional to the deforming force or load. Under these conditions the object returns to its original shape and size upon removal of the load. Elastic behaviour of solids according to Hooke's law can be explained by the fact that small displacements of their constituent molecules, atoms, or ions from normal positions is also proportional to the force that causes the displacement.
The deforming force may be applied to a solid by stretching, compressing, squeezing, bending, or twisting. Thus a metal wire exhibits elastic behaviour according to Hooke's law because the small increase in its length when stretched by an applied force doubles each time the force is doubled. Mathematically Hooke's law states that the applied force F equals a constant k times the displacement or change in length x, or $F = kx$. The value of k depends not only on the kind of elastic material under consideration but also on its dimensions and shape. At relatively large values of applied force, the deformation of the elastic material is often larger than expected on the basis of Hooke's law, even though the material remains elastic and returns to its original shape and size after removal of the force. Hooke's law describes the elastic properties of materials only in the range in which the force and displacement are proportional. Sometimes Hooke's law is formulated as $F = -kx$. In this expression F no longer means the applied force but rather the equal and oppositely directed restoring force that causes elastic materials to return to their original dimensions.
Hooke's law may also be expressed in terms of stress and strain. Stress is the force on unit areas within a material that develops as a result of the externally applied force. Strain is the relative deformation produced by stress. For relatively small stresses, stress is proportional to strain.

Robert Hooke, a contemporary of Newton, tried to define a set of mathematical laws to predict the behaviour of forces directed towards the centre of something ('central forces'), (e.g., planetary gravity.) He examined central forces that were inversely proportional to the distance between two objects. This linear relationship does not hold true in light of our present knowledge of planetary gravitation, but works well for springs. Hooke's Law, as commonly used, states that the force a spring exerts on a body is directly proportional to the displacement of the system (extension of the spring). That is, $\boldsymbol{F = -k\,x}$, where $\boldsymbol{F}$ is the force exerted, $\boldsymbol{x}$ is the extension of the spring, and k is the proportionality (or spring) constant that varies from spring to spring. Other forms, applicable to collapsing balls and other systems, are $\boldsymbol{F = -k\,x^2}$, $\boldsymbol{F = -k\,x^{3/2}}$ and other powers of x.

Shear modulus
numerical constant that describes the elastic properties of a solid under the application of transverse internal forces such as arise, for example, in torsion, as in twisting a metal pipe about its lengthwise axis. Within such a material any small cubic volume is slightly distorted in such a way that two of its faces slide parallel to each other a small distance and two other faces change from squares to diamond shapes. The shear modulus is a measure of the ability of a material to resist transverse deformations and is a valid index of elastic behaviour only

for small deformations, after which the material is able to return to its original configuration. Large shearing forces lead to flow and permanent deformation or fracture. The shear modulus is also known as the rigidity.

Mathematically the shear modulus is equal to the quotient of the shear stress divided by the shear strain. The shear stress, in turn, is equal to the shearing force F divided by the area A parallel to and in which it is applied, or F/A. The shear strain or relative deformation is a measure of the change in geometry and in this case is expressed by the trigonometric function, tangent (tan) of the angle θ (theta), which denotes the amount of change in the 90°, or right, angles of the minute representative cubic volume of the unstrained material. Mathematically, shear strain is expressed as tan θ or its equivalent, by definition, x/y. The shear modulus itself may be expressed mathematically as
shear modulus = shear strain shear stress = x/y F/A.
This equation is a specific form of Hooke’s Law of elasticity. Because the denominator is a ratio and thus dimensionless, the dimensions of the shear modulus are those of force per unit area. In the English system the shear modulus may be expressed in units of pounds per square inch (usually abbreviated to psi); the common SI units are newtons per square metre (N/m^2). The value of the shear modulus for aluminium is about 3.5×10^6 psi, or 2.4×10^{10} N/m^2. By comparison, steel under shear stress is more than three times as rigid as aluminium.

Snell’s Law
Seismic waves change speed within the earth and follow curved paths, due to changes in the elastic properties of the material inside the earth, in accordance with Snell's Law. (In this set of activities, we keep things simple by considering only waves travelling through the earth's mantle, although the same principles apply to waves travelling through the core.) Just as changes in the optical density of a material will bend the paths of light waves passing through the material, changes in the elastic properties of the material inside the earth bends the paths of mechanical waves, such as seismic waves.
Within the earth's crust and mantle, material becomes increasingly dense and rigid at greater depths. These changes lead to an increase in speed of both compressional and shear waves as they travel through increasing depths. With the increase in speed comes a change in direction. The relationship among the angle of incidence (i), the velocities of the wave in each layer of material (v1 and v2), and the angle of refraction (r) is governed by Snell's Law:

$$\mathbf{\sin i/v1 = \sin r/v2}$$

(This may also be expressed in terms of the index of refraction, n = sin i/sin r.)
When a seismic wave encounters a boundary between two types of material with different values for the index of refraction, the wave may bend so that it is travelling along the boundary, rather than into the new material. This happens only for waves arriving from a particular angle of incidence, called the critical angle. When the angle of incidence is greater than the critical angle, the seismic wave is reflected off the boundary, back into the first layer of material, rather than passing into the second layer. This is called internal reflection. Seismologists can learn about the layers in the earth by studying these reflected rays.

Transfer function
The engineering terminology for one use of the Fourier transforms. By breaking up a wave pulse into its frequency spectrum

$$f_\nu = \mathrm{F}(\nu)\, e^{2\pi i \nu t}$$

the entire signal can be written as a sum of contributions from each frequency,

$$f(t) = \int_{-\infty}^{\infty} f_\nu d\nu = \int_{-\infty}^{\infty} F(\nu) e^{2\pi\nu t} d\nu$$

If the signal is modified in some way, it will become

$$g_\nu(t) = \phi(\nu) f_\nu(t) = \phi(\nu) F(\nu) e^{2\pi i \nu t}$$

$$g(t) = \int_{-\infty}^{\infty} g_\nu(t) dt = \int_{-\infty}^{\infty} \phi(\nu) F(\nu) e^{2\pi i \nu t} d\nu$$

where ϕ (ν) is known as the "transfer fuction". Fourier transforming ϕ and F,

$$\phi(\nu) = \int_{-\infty}^{\infty} \Phi(t) e^{-2\pi i \nu t} dt$$

$$F(\nu) = \int_{-\infty}^{\infty} f(t) e^{-2\pi i \nu t} dt$$

From the convolution theorem,

$$g(t) = f(t) * \Phi(t) = \int_{-\infty}^{\infty} f(t) \Phi(t - \tau) d\tau$$

Appendix C
Dictionary

abscissas
The co-ordinate representing the position of a point along a line perpendicular to the y-axis in a plane Cartesian co-ordinate system.

analytic
5.Logic. Following necessarily; tautologous: an analytic truth.
6.Mathematics.
a.Using, subjected to, or capable of being subjected to a methodology involving algebra and calculus.
b.Proving a known truth by reasoning from that which is to be proved.

algorithm
A step-by-step problem-solving procedure, especially an established, recursive computational procedure for solving a problem in a finite number of steps.

ambient
Surrounding; encircling: ambient sound; ambient air.

argument
Mathematics.
a.The independent variable of a function.
b.The amplitude of a complex number.
Computer Science. A value used to evaluate a procedure or subroutine.

bias
A statistical sampling or testing error caused by systematically favouring some outcomes over others.

confidence interval
A statistical range with a specified probability that a given parameter lies within the range.

chi-square
A test statistic that is calculated as the sum of the squares of observed values minus expected values divided by the expected values.

coil
Electricity.
a.A wound spiral of two or more turns of insulated wire, used to introduce inductance into a circuit.
b.Any of various devices of which such a spiral is the major component.

correlation
2.Statistics. The simultaneous change in value of two numerically valued random variables: the positive correlation between cigarette smoking and the incidence of lung cancer; the negative correlation between age and normal vision.

covariance
A statistical measure of the variance of two random variables that are observed or measured in the same mean time period. This measure is equal to the product of the deviations of corresponding values of the two variables for their respective means.

discrete Fourier transform

<mathematics> (DFT) A Fourier transform, specialised to the case where the abscissas are integers.

The DFT is central to many kinds of signal processing, including the analysis and compression of video and sound information.

A common implementation of the DFT is the Fast Fourier Transform (FFT).

entropy

- A measure of the disorder or randomness in a closed system.
- A measure of the number of bits necessary to transmit a message as a function of the probability that the message will consist of a specific set of symbols.
- A hypothetical tendency for all matter and energy in the universe to evolve toward a state of inert uniformity.
- Inevitable and steady deterioration of a system or society.

evaluate

Mathematics. To calculate the numerical value of; express numerically.

Fourier transform

<mathematics> A technique for expressing a waveform as a weighted sum of sines and cosines.

Computers generally rely on the version known as discrete Fourier transform.

frequency

2.Mathematics & Physics. The number of times a specified phenomenon occurs within a specified interval, as:

a.The number of repetitions of a complete sequence of values of a periodic function per unit variation of an independent variable.

b.The number of complete cycles of a periodic process occurring per unit time.

c.The number of repetitions per unit time of a complete waveform, as of an electric current.

3.Statistics.

a.The number of measurements in an interval of a frequency distribution.

b.The ratio of the number of times an event occurs in a series of trials of a chance experiment to the number of trials of the experiment performed.

geophone

An electronic receiver designed to pick up seismic vibrations.

gimbals

A device consisting of two rings mounted on axes at right angles to each other so that an object, such as a ship's compass, will remain suspended in a horizontal plane between them regardless of any motion of its support. Often used in the plural.

infrasonic

having frequencies below those of audible sound

inertia

Physics. The tendency of a body to resist acceleration; the tendency of a body at rest to remain at rest or of a body in motion to stay in motion in a straight line unless acted on by an outside force.

MATLAB

MATrix LABoratory. A high performance language for technical computing. Integrates computation, visualisation and programming

polarisation

1.The production or condition of polarity, as:

a.A process or state in which rays of light exhibit different properties in different directions, especially the state in which all the vibration takes place in one plane.

2.A concentration, as of groups, forces, or interests, about two conflicting or contrasting positions.

parametric

1.Mathematics.

a.A constant in an equation that varies in other equations of the same general form, especially such a constant in the equation of a curve or surface that can be varied to represent a family of curves or surfaces.

b.One of a set of independent variables that express the coordinates of a point.

2.

a.One of a set of measurable factors, such as temperature and pressure, that define a system and determine its behaviour and are varied in an experiment.

b.Usage Problem. A factor that restricts what is possible or what results: "all the parameters of shelter where people will live, what mode of housing they will choose, and how they will pay for it" (New York).

c.A factor that determines a range of variations; a boundary: an experimental school that keeps expanding the parameters of its curriculum.

3. Statistics. A quantity, such as a mean, that is calculated from data and describes a population.

resonance

2.Physics. The increase in amplitude of oscillation of an electric or mechanical system exposed to a periodic force whose frequency is equal or very close to the natural undamped frequency of the system.

3.Acoustics. Intensification and prolongation of sound, especially of a musical tone, produced by sympathetic vibration.

standard deviation

A statistic used as a measure of the dispersion or variation in a distribution, equal to the square root of the arithmetic mean of the squares of the deviations from the arithmetic mean.

statement

Computer Science. An elementary instruction in a source language.

spectrum

1.Physics. The distribution of a characteristic of a physical system or phenomenon, especially:

a.The distribution of energy emitted by a radiant source, as by an incandescent body, arranged in order of wavelengths.

b.The distribution of atomic or subatomic particles in a system, as in a magnetically resolved molecular beam, arranged in order of masses.

2.A graphic or photographic representation of such a distribution.

3.

a.A range of values of a quantity or set of related quantities.

spectral analysis
seeks to describe the frequency content of a signal, random process, or system, based on a finite set of data

string
Computer Science. A set of consecutive characters treated by a computer as a single item.

taper
1.To become gradually narrower or thinner toward one end.
2.To diminish or lessen gradually.

variance
Statistics. The square of the standard deviation.

Appendix D
Matlabcommands

- COLORMAP Colour look-up table.
  COLORMAP(MAP) sets the current figure's colourmap to MAP.
  COLORMAP('default') sets the current figure's colormap to
  the root's default, whose setting is JET.
  MAP = COLORMAP retrieves the current colourmap. The values
  are in the range from 0 to 1.

  A colour map matrix may have any number of rows, but it must have
  exactly 3 columns. Each row is interpreted as a colour, with the
  first element specifying the intensity of red light, the second
  green, and the third blue. Colour intensity can be specified on the
  interval 0.0 to 1.0.
  For example, [0 0 0] is black, [1 1 1] is white,
  [1 0 0] is pure red, [.5 .5 .5] is gray, and
  [127/255 1 212/255] is aquamarine.

  Graphics objects that use pseudocolour -- SURFACE and PATCH objects,
  which are created by the functions MESH, SURF, and PCOLOR -- map
  a colour matrix, C, whose values are in the range [Cmin, Cmax],
  to an array of indices, k, in the range [1, m].
  The values of Cmin and Cmax are either min(min(C)) and max(max(C)),
  or are specified by CAXIS. The mapping is linear, with Cmin
  mapping to index 1 and Cmax mapping to index m. The indices are
  then used with the colourmap to determine the colour associated
  with each matrix element. See CAXIS for details.

  Type HELP GRAPH3D to see a number of useful colourmaps.

  COLORMAP is an M-file that sets the Colourmap property of the current
  figure.

  See also HSV, CAXIS, SPINMAP, BRIGHTEN, RGBPLOT, FIGURE.

- FOR Repeat statements a specific number of times.
  The general form of a FOR statement is:
  FOR variable = expr, statement, ..., statement END

  The columns of the expression are stored one at a time in
  the variable and then the following statements, up to the
  END, are executed.

- FIND Find indices of nonzero elements.

I = FIND(X) returns the indices of the vector X that are
non-zero. For example, I = FIND(A>100), returns the indices
of A where A is greater than 100. See RELOP.

[I,J] = FIND(X) returns the row and column indices of
the nonzero entries in the matrix X. This is often used
with sparse matrices.

[I,J,V] = FIND(X) also returns a vector containing the
nonzero entries in X. Note that find(X) and find(X~=0)
will produce the same I and J, but the latter will produce
a V with all 1's.

See also SPARSE, IND2SUB.

- FILTER One-dimensional digital filter.
  Y = FILTER(B,A,X) filters the data in vector X with the
  filter described by vectors A and B to create the filtered
  data Y. The filter is a "Direct Form II Transposed"
  implementation of the standard difference equation:

  a(1)*y(n) = b(1)*x(n) + b(2)*x(n-1) + ... + b(nb+1)*x(n-nb)
  - a(2)*y(n-1) - ... - a(na+1)*y(n-na)

  If a(1) is not equal to 1, FILTER normalises the filter
  coefficients by a(1).

  When X is a matrix, FILTER operates on the columns of X. When X
  is an N-D array, FILTER operates along the first non-singleton
  dimension.

- IF is a statement condition.
  The general form of the IF statement is

```
IF expression
  statements
ELSEIF expression
  statements
ELSE
  statements
END
```

  The expression is usually of the form expr rop expr where
  rop is ==, <, >, <=, >=, or ~=.

- SORT Sort in ascending order.
  For vectors, SORT(X) sorts the elements of X in ascending order
  For matrices, SORT(X) sorts each column of X in ascending order.
  For N-D arrays, SORT(X) sorts the along the first non-singleton
  dimension of X. When X is a cell array of strings, SORT(X) sorts

the strings in ASCII dictionary order.

SORT(X,DIM) sorts along the dimension DIM.

[Y,I] = SORT(X) also returns an index matrix I. If X is a
vector, then Y = X(I). If X is an m-by-n matrix, then
for j = 1:n, Y(:,j) = X(I(:,j),j); end

When X is complex, the elements are sorted by ABS(X). Complex
matches are further sorted by ANGLE(X).

```
Example: If X = [3 7 5
                 0 4 2]

then sort(X,1) is [0 4 2  and sort(X,2) is [3 5 7
                   3 7 5]                   0 2 4];
```

See also SORTROWS, MIN, MAX, MEAN, MEDIAN.

Overloaded methods
help cell/sort.m

- SPRINTF Write formatted data to string.
[S,ERRMSG] = SPRINTF(FORMAT,A,...) formats the data in the real part of matrix A (and in any additional matrix arguments), under control of the specified FORMAT string, and returns it in the MATLAB string variable S. ERRMSG is an optional output argument that returns an error message string if an error occurred or an empty matrix if an error did not occur. SPRINTF is the same as FPRINTF except that it returns the data in a MATLAB string variable rather than writing it to a file.

FORMAT is a string containing C language conversion specifications. Conversion specifications involve the character %, optional flags, optional width and precision fields, optional subtype specifier, and conversion characters d, i, o, u, x, X, f, e, E, g, G, c, and s.
See the Language Reference Guide or a C manual for complete details.

The special formats \n,\r,\t,\b,\f can be used to produce linefeed, carriage return, tab, backspace, and formfeed characters respectively. Use \\ to produce a backslash character and %% to produce the percent character.

SPRINTF behaves like ANSI C with certain exceptions and extensions.
These include:
1. The following non-standard subtype specifiers are supported for conversion characters o, u, x, and X.
   t - The underlying C datatype is a float rather than an unsigned integer.

b - The underlying C datatype is a double rather than an unsigned integer.
For example, to print out in hex a double value use a format like '%bx'.

2. SPRINTF is "vectorized" for the case when A is nonscalar. The format string is recycled through the elements of A (columnwise) until all the elements are used up. It is then recycled in a similar manner through any additional matrix arguments.

Examples

```
sprintf('%0.5g',(1+sqrt(5))/2)      1.618
sprintf('%0.5g',1/eps)              4.5036e+15
sprintf('%15.5f',1/eps)             4503599627370496.00000
sprintf('%d',round(pi))             3
sprintf('%s','hello')               hello
sprintf('The array is %dx%d.',2,3)  The array is 2x3.
sprintf('\n') is the line termination character on all platforms.
```

- INTERP Resample data at a higher rate using lowpass interpolation.
  Y = INTERP(X,R) resamples the sequence in vector X at R times the original sample rate. The resulting resampled vector Y is R times longer, LENGTH(Y) = R*LENGTH(X).

  A symmetric filter, B, allows the original data to pass through unchanged and interpolates between so that the mean square error between them and their ideal values is minimized.
  Y = INTERP(X,R,L,ALPHA) allows specification of arguments L and ALPHA which otherwise default to 4 and .5 respectively. 2*L is the number of original sample values used to perform the interpolation. Ideally L should be less than or equal to 10. The length of B is 2*L*R+1. The signal is assumed to be band limited with cutoff frequency 0 < ALPHA <= 1.0.
  [Y,B] = INTERP(X,R,L,ALPHA) returns the coefficients of the interpolation filter B. See also DECIMATE.

- INTERP1 1-D interpolation (table lookup).
  YI = INTERP1(X,Y,XI) interpolates to find YI, the values of the underlying function Y at the points in the vector XI. The vector X specifies the points at which the data Y is given. If Y is a matrix, then the interpolation is performed for each column of Y and YI will be length(XI)-by-size(Y,2). Out of range values are returned as NaN.

  YI = INTERP1(Y,XI) assumes X = 1:N, where N is the length(Y) for vector Y or SIZE(Y,1) for matrix Y.

  Interpolation is the same operation as "table lookup". Described in "table lookup" terms, the "table" is [X,Y] and INTERP1 "looks-up"

```
the elements of XI in X, and, based upon their location, returns
values YI interpolated within the elements of Y.

YI = INTERP1(X,Y,XI,'method') specifies alternate methods.
The default is linear interpolation.  Available methods are:

  'nearest' - nearest neighbor interpolation
  'linear'  - linear interpolation
  'spline'  - cubic spline interpolation
  'cubic'   - cubic interpolation

All the interpolation methods require that X be monotonic. X can be
non-uniformly spaced.  For faster interpolation when X is equally
spaced and monotonic, use the methods '*linear', '*cubic', '*nearest'.
For faster linear interpolation when X is non-uniformly spaced
see INTERP1Q.

For example, generate a coarse sine curve and interpolate over a
finer abscissa:
    x = 0:10; y = sin(x); xi = 0:.25:10;
    yi = interp1(x,y,xi); plot(x,y,'o',xi,yi)

See also INTERP1Q, INTERPFT, SPLINE, INTERP2, INTERP3, INTERPN
```

Appendix E

- **t02s_tomgetspecdata_template1.m**

```
clf
clear

  fileh(1,:)  = '../spectra/FILEH01';
  fileh(2,:)  = '../spectra/FILEH02';
  fileh(3,:)  = '../spectra/FILEH03';
  fileh(4,:)  = '../spectra/FILEH04';
  fileh(5,:)  = '../spectra/FILEH05';
  fileh(6,:)  = '../spectra/FILEH06';
  fileh(7,:)  = '../spectra/FILEH07';
  fileh(8,:)  = '../spectra/FILEH08';
  fileh(9,:)  = '../spectra/FILEH09';
  fileh(10,:) = '../spectra/FILEH10';
  fileh(11,:) = '../spectra/FILEH11';
  fileh(12,:) = '../spectra/FILEH12';
  fileh(13,:) = '../spectra/FILEH13';

  filev(1,:)  = '../spectra/FILEV01';
  filev(2,:)  = '../spectra/FILEV02';
  filev(3,:)  = '../spectra/FILEV03';
  filev(4,:)  = '../spectra/FILEV04';
  filev(5,:)  = '../spectra/FILEV05';
  filev(6,:)  = '../spectra/FILEV06';
  filev(7,:)  = '../spectra/FILEV07';
  filev(8,:)  = '../spectra/FILEV08';
  filev(9,:)  = '../spectra/FILEV09';
  filev(10,:) = '../spectra/FILEV10';
  filev(11,:) = '../spectra/FILEV11';
  filev(12,:) = '../spectra/FILEV12';
  filev(13,:) = '../spectra/FILEV13';

  dbtime(1,:)  = DBTIME01;
  dbtime(2,:)  = DBTIME02;
  dbtime(3,:)  = DBTIME03;
  dbtime(4,:)  = DBTIME04;
  dbtime(5,:)  = DBTIME05;
  dbtime(6,:)  = DBTIME06;
  dbtime(7,:)  = DBTIME07;
  dbtime(8,:)  = DBTIME08;
  dbtime(9,:)  = DBTIME09;
  dbtime(10,:) = DBTIME10;
  dbtime(11,:) = DBTIME11;
  dbtime(12,:) = DBTIME12;
  dbtime(13,:) = DBTIME13;

  nfftC=32768;
  [mag,phase,fout,kend]=TF54000(nfftC/2);
%
  for ichan=1:2
    if ichan==1
      orientation='z';
%
```

```
%  The data files for day 62 have already been uncompressed
%
      fid=fopen('WHERE1/Z_DBFILE1');
      C1=fread(fid,'float');
      fclose(fid);
      if (doit == 1)

        fid=fopen('WHERE2/Z_DBFILE2');
        C11=fread(fid,'float');
        fclose(fid);
        C1=[C1;C11];

      end
%
    elseif ichan==2
      orientation='n';
%
      fid=fopen('WHERE1/1_DBFILE1');
      C2=fread(fid,'float');
      fclose(fid);
%
      fid=fopen('WHERE1/2_DBFILE1');
      C3=fread(fid,'float');
      fclose(fid);

      if (doit == 1)

        fid=fopen('WHERE2/1_DBFILE2');
        C21=fread(fid,'float');
        fclose(fid);
        C2=[C2;C21];

        fid=fopen('WHERE2/2_DBFILE2');
        C31=fread(fid,'float');
        fclose(fid);
        C3=[C3;C31];

      end
%
      C1=sqrt(C2.^2+C3.^2);
%
    end

    kmm=length(C1);
    kmm=kmm/131072;
    kmm=fix(kmm);
%
    for ichunk=1:kmm
      ist=(ichunk-1)*131072+1;
      C2=C1(ist:ist+131072-1);
      FC=20;
      nfftC=32768;
      windowC=hanning(nfftC);
      noverlapC=0.75*nfftC;
      dflagC='linear';
      [PxxC,fC]=psd(C2,nfftC,FC,windowC,noverlapC,dflagC);
      NN=length(fC);
      PxxC=PxxC(2:NN-1);
      fC=fC(2:NN-1);
%
%  Amplitude adjustments
```

```
%
      sintr=0.05;
      NN=length(PxxC);
%
%  Multiply by delt to get correct root(Hz) behaviour
%   (See SPECtests1norm)
%
      PxxC=PxxC.*sintr;
%
%  Apply transfer function (includes conversion to accel and cal factor)
%
      [mag,phase,fout,kend]=TF54000(nfftC/2);
      PxxC = PxxC .* (mag.*10^(-9)).^2;
      PxxC=PxxC(1:kend);
      fC=fC(1:kend);
%
%  Smooth the spectra
%
      NPD=64;
      NN=length(PxxC);
      [ff,PP]=smooth_spec(fC,PxxC,NPD);
      A=[ff,10*log10(PP)];
%
%  f should be the same for all spectra; SAdb should be saved for plotting
%  in waterfall format
%
      f=A(:,1);
      SAdb=A(:,2);
%
%  Plot the data (just for single spectra)
%
      if ichan==1
        fip = fopen(filev(ichunk,:),'w+');
        fprintf (fip,' %f \n',dbtime(ichunk,:));
        fprintf (fip,' %f ',SAdb);
        fprintf (fip,'\n');
        fclose (fip);

%        semilogx(f,SAdb,'b-')
%        hold on
      elseif ichan==2
%        semilogx(f,SAdb,'r-')
        fip = fopen(fileh(ichunk,:),'w+');
        fprintf (fip,' %f \n',dbtime(ichunk,:));
        fprintf (fip,' %f ',SAdb);
        fprintf (fip,'\n');
        fclose (fip);
%        hold on
      end
    end
  end %  of chunk loop
%
%  Plot Legend
%
%  shift=50;
%  x=[1.5*10^(-3) 1.5*10^(-3) 0.8*10^(-1) 0.8*10^(-1)];
%  y=[-156+shift -143+shift -143+shift -156+shift];
%  patch(x,y,'w')
%  text(2*10^(-3),-146+shift,'OSN-VERT     ');
%  text(2*10^(-3),-152+shift,'OSN-HORI(rms) ');
%  plot([0.3*10^(-1) 0.6*10^(-1)],[-146+shift -146+shift],'b-')
```

```
%  plot([0.3*10^(-1) 0.6*10^(-1)],[-152+shift -152+shift],'r-')
%
% draw the low and high Noise curves from Peterson
%
%  for m=1:2
%    if m==1
%      eval(['load nlnm_avd ;'])
%      eval(['NM=nlnm_avd;']);
%    elseif m==2
%      eval(['load nhnm_avd ;'])
%      eval(['NM=nhnm_avd;']);
%    end
%    km=1:length(NM);
%    tm  =NM(km,1);        % periods(sec)
%    fm  =NM(km,2);        % frequency(Hz)
%    acc =NM(km,3);     % db refered to acceleration
%    vel =NM(km,4);     % db refered to velocity
%    dis =NM(km,5);     % db refered to displacement
%    semilogx(fm,acc,'k--');
%  end
%  hold on
%
%  End of Script
```

- **t03s_irenetest.m**

```
clf
clear

  fileh(1,:)  = '../spectra/test01H.spec';
      ect.



  nfftC=32768;
  [mag,phase,fout,kend]=TF54000(nfftC/2);
%
  for ichan=1:2
    if ichan==1
      orientation='z';
%
%  The data files for day 62 have already been uncompressed
%
      fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BHZ');
      C1=fread(fid,131072,'float');
      fclose(fid);
%
    elseif ichan==2
      orientation='n';
%
      fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH1');
      C2=fread(fid,131072,'float');
      fclose(fid);
%
      fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH2');
      C3=fread(fid,131072,'float');
      fclose(fid);
%
      C1=sqrt(C2.^2+C3.^2);
%
```

```
    end

    kmm=length(C1);
    kmm=kmm/131072;
    kmm=fix(kmm);
%
    for ichunk=1:kmm
      ist=(ichunk-1)*131072+1;
      C2=C1(ist:ist+131072-1);
      FC=20;
      nfftC=32768;
      windowC=hanning(nfftC);
      noverlapC=0.75*nfftC;
      dflagC='linear';
      [PxxC,fC]=psd(C2,nfftC,FC,windowC,noverlapC,dflagC);
      NN=length(fC);
      PxxC=PxxC(2:NN-1);
      fC=fC(2:NN-1);
%
%  Amplitude adjustments
%
      sintr=0.05;
      NN=length(PxxC);
%
%  Multiply by delt to get correct root(Hz) behaviour
%   (See SPECtests1norm)
%
      PxxC=PxxC.*sintr;
%
%  Apply transfer function (includes conversion to accel and cal factor)
%
      [mag,phase,fout,kend]=TF54000(nfftC/2);
      PxxC = PxxC .* (mag.*10^(-9)).^2;
      PxxC=PxxC(1:kend);
      fC=fC(1:kend);
%
%  Smooth the spectra
%
      NPD=64;
      NN=length(PxxC);
      [ff,PP]=smooth_spec(fC,PxxC,NPD);
      A=[ff,10*log10(PP)];
%
%  f should be the same for all spectra; SAdb should be saved for plotting
%  in waterfall format
%
      f=A(:,1);
      SAdb=A(:,2);
%
%  Plot the data (just for single spectra)
%
      if ichan==1
        fip = fopen(filev(ichunk,:),'w+');
%        fprintf (fip,' %f \n',dbtime(ichunk,:));
        fprintf (fip,' %f ',SAdb);
        fprintf (fip,'\n');
        fclose (fip);

        semilogx(f,SAdb,'b-')
        hold on
      elseif ichan==2
```

```
        semilogx(f,SAdb,'r-')
        fip = fopen(fileh(ichunk,:),'w+');
%        fprintf (fip,' %f \n',dbtime(ichunk,:));
        fprintf (fip,' %f ',SAdb);
        fprintf (fip,'\n');
        fclose (fip);
        hold on
      end
    end
  end %  of chunk loop
%
%  Plot Legend
%
  shift=50;
  x=[1.5*10^(-3) 1.5*10^(-3) 0.8*10^(-1) 0.8*10^(-1)];
  y=[-156+shift -143+shift -143+shift -156+shift];
  patch(x,y,'w')
  text(2*10^(-3),-146+shift,'OSN-VERT     ');
  text(2*10^(-3),-152+shift,'OSN-HORI(rms) ');
  plot([0.3*10^(-1) 0.6*10^(-1)],[-146+shift -146+shift],'b-')
  plot([0.3*10^(-1) 0.6*10^(-1)],[-152+shift -152+shift],'r-')
%
% draw the low and high Noise curves from Peterson
%
  for m=1:2
    if m==1
      eval(['load ../matlab/nlnm_avd ;'])
      eval(['NM=nlnm_avd;']);
    elseif m==2
      eval(['load ../matlab/nhnm_avd ;'])
      eval(['NM=nhnm_avd;']);
    end
    km=1:length(NM);
    tm  =NM(km,1);        % periods(sec)
    fm  =NM(km,2);        % frequency(Hz)
    acc =NM(km,3);      % db refered to acceleration
    vel =NM(km,4);      % db refered to velocity
    dis =NM(km,5);      % db refered to displacement
    semilogx(fm,acc,'k--');
  end
  hold on
%
%  End of Script
```

- **t04_puremfile.m**

```
clf
clear

  fileh(1,:)  = '../spectra/test01H.spec';
      ect.

nfftC=32768;
[mag,phase,fout,kend]=TF54000(nfftC/2);
  for ichan=1:2
    if ichan==1
      orientation='z';

      fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BHZ');
```

```
    C1=fread(fid,131072,'float');
    fclose(fid);

  elseif ichan==2
    orientation='n';

    fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH1');
    C2=fread(fid,131072,'float');
    fclose(fid);

    fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH2');
    C3=fread(fid,131072,'float');
    fclose(fid);

    C1=sqrt(C2.^2+C3.^2);

  end

  kmm=length(C1);
  kmm=kmm/131072;
  kmm=fix(kmm);

  for ichunk=1:kmm
    ist=(ichunk-1)*131072+1;
    C2=C1(ist:ist+131072-1);
    FC=20;
    nfftC=32768;
    windowC=hanning(nfftC);
    noverlapC=0.75*nfftC;
    dflagC='linear';
    [PxxC,fC]=psd(C2,nfftC,FC,windowC,noverlapC,dflagC);
    NN=length(fC);
    PxxC=PxxC(2:NN-1);
    fC=fC(2:NN-1);

    sintr=0.05;
    NN=length(PxxC);

    PxxC=PxxC.*sintr;

    [mag,phase,fout,kend]=TF54000(nfftC/2);
    PxxC = PxxC .* (mag.*10^(-9)).^2;
    PxxC=PxxC(1:kend);
    fC=fC(1:kend);

    NPD=64;
    NN=length(PxxC);
    [ff,PP]=smooth_spec(fC,PxxC,NPD);
    A=[ff,10*log10(PP)];

    f=A(:,1);
    SAdb=A(:,2);

    if ichan==1
      fip = fopen(filev(ichunk,:),'w+');
      fprintf (fip,' %f ',SAdb);
      fprintf (fip,'\n');
      fclose (fip);

      semilogx(f,SAdb,'b-')
      hold on
```

```
      elseif ichan==2
        semilogx(f,SAdb,'r-')
        fip = fopen(fileh(ichunk,:),'w+');
        fprintf (fip,' %f ',SAdb);
        fprintf (fip,'\n');
        fclose (fip);
        hold on
      end
    end
  end

  shift=50;
  x=[1.5*10^(-3) 1.5*10^(-3) 0.8*10^(-1) 0.8*10^(-1)];
  y=[-156+shift -143+shift -143+shift -156+shift];
  patch(x,y,'w')
  text(2*10^(-3),-146+shift,'OSN-VERT     ');
  text(2*10^(-3),-152+shift,'OSN-HORI(rms) ');
  plot([0.3*10^(-1) 0.6*10^(-1)],[-146+shift -146+shift],'b-')
  plot([0.3*10^(-1) 0.6*10^(-1)],[-152+shift -152+shift],'r-')

  for m=1:2
    if m==1
      eval(['load ../matlab/nlnm_avd ;'])
      eval(['NM=nlnm_avd;']);
    elseif m==2
      eval(['load ../matlab/nhnm_avd ;'])
      eval(['NM=nhnm_avd;']);
    end
    km=1:length(NM);
    tm  =NM(km,1);        % periods(sec)
    fm  =NM(km,2);        % frequency(Hz)
    acc =NM(km,3);      % db refered to acceleration
    vel =NM(km,4);      % db refered to velocity
    dis =NM(km,5);      % db refered to displacement
    semilogx(fm,acc,'k--');
  end
  hold on
```

- **t05_commentfile.m**

```
clf
clear
%
  fileh(1,:)  = '../spectra/test01H.spec';
      ect.

nfftC=32768;
[mag,phase,fout,kend]=TF54000(nfftC/2);
%
  for ichan=1:2
    if ichan==1
      orientation='z';
%-------------------------------------------------------------------
%     FOR    Repeat statements a specific number of times.
%   The general form of a FOR statement is:
%
%      FOR variable = expr, statement, ..., statement END
%
%   The columns of the expression are stored one at a time in
```

```
%   the variable and then the following statements, up to the
%   END, are executed.
%-------------------------------------------------------------------
%     IF is a statement condition.
%   The general form of the IF statement is
%
%      IF expression
%        statements
%      ELSEIF expression
%        statements
%      ELSE
%        statements
%      END
%
%     The expression is usually of the form expr rop expr where
%    rop is ==, <, >, <=, >=, or ~=.
%-------------------------------------------------------------------
%
%  The data files for day 62 have already been uncompressed
%-------------------------------------------------------------------
      fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BHZ');
      C1=fread(fid,131072,'float');
      fclose(fid);
      if (doit == 1)

        fid=fopen('../osn1_compressed/1998/096/Z_DBFILE2');
        C11=fread(fid,'float');
        fclose(fid);
        C1=[C1;C11];

      end
%-------------------------------------------------------------------
%     FOPEN  Open file.
%   FID = FOPEN(FILENAME,PERMISSION) opens the specified file with
%   the specified PERMISSION.
%   If the open is successful, FID gets a scalar MATLAB integer, the
%   file identifier, to be used as the first argument to other FileIO
%   routines. If the open was not successful, -1 is returned for FID.
%-------------------------------------------------------------------
%     FREAD  Read binary data from file.
%   [A, COUNT] = FREAD(FID,SIZE,PRECISION) reads binary data from the
%   specified file and writes it into matrix A.  Optional output
%   argument COUNT returns the number of elements successfully read.
%
%   FID is an integer file identifier obtained from FOPEN.
%
%   The precision argument controls the number of bits read for each value
%   and the interpretation of those bits as character, integer or floating
%   point values.
%   'float32' 'real*4'         floating point, 32 bits.
%-------------------------------------------------------------------
%      FCLOSE Close file.
%   ST = FCLOSE(FID) closes the file with file identifier FID,
%   which is an integer obtained from an earlier FOPEN.  FCLOSE
%   returns 0 if successful and -1 if not.
%
%   ST = FCLOSE('all') closes all open files, except 0, 1 and 2.
%-------------------------------------------------------------------
%      VERTCAT Vertical concatenation.
%   [A;B] is the vertical concatenation of matrices A and B.
%   A and B must have the same number of columns.
```

```
%-------------------------------------------------------------------
    elseif ichan==2
      orientation='n';
%
      fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH1');
      C2=fread(fid,131072,'float');
      fclose(fid);
%
      fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH2');
      C3=fread(fid,131072,'float');
      fclose(fid);

      if (doit == 1)

        fid=fopen('../osn1_compressed/1998/096/1_DBFILE2');
        C21=fread(fid,'float');
        fclose(fid);
        C2=[C2;C21];

        fid=fopen('../osn1_compressed/1998/096/2_DBFILE2');
        C31=fread(fid,'float');
        fclose(fid);
        C3=[C3;C31];

      end
%
      C1=sqrt(C2.^2+C3.^2);
%
    end

    kmm=length(C1);
    kmm=kmm/131072;
    kmm=fix(kmm);
%-------------------------------------------------------------------
%     LENGTH   Length of vector.
%   LENGTH(X) returns the length of vector X.  It is equivalent
%   to MAX(SIZE(X)) for non-empty arrays and 0 for empty ones.
%-------------------------------------------------------------------
%     FIX    Round towards zero.
%   FIX(X) rounds the elements of X to the nearest integers
%   towards zero.
%-------------------------------------------------------------------
    for ichunk=1:kmm
      ist=(ichunk-1)*131072+1;
      C2=C1(ist:ist+131072-1);
      FC=20;
      nfftC=32768;
      windowC=hanning(nfftC);
%--------------------------------------------------------------------------
%HANNING Hanning window.
%   HANNING(N) returns the N-point symmetric Hanning window
%   in a column vector.  Note that the first and last zero-
%   weighted window samples are not included.
%
%   HANNING(N,'symmetric') returns the same result as HANNING(N).
%
%   HANNING(N,'periodic') returns the N-point periodic Hanning
%   window, and includes the first zero-weighted window sample.
%--------------------------------------------------------------------------
      noverlapC=0.75*nfftC;
      dflagC='linear';
```

```
      [PxxC,fC]=psd(C2,nfftC,FC,windowC,noverlapC,dflagC);
      NN=length(fC);
      PxxC=PxxC(2:NN-1);
      fC=fC(2:NN-1);
%---------------------------------------------------------------------------
%  Amplitude adjustments
%---------------------------------------------------------------------------
      sintr=0.05;
      NN=length(PxxC);
%---------------------------------------------------------------------------
%  Multiply by delt to get correct root(Hz) behaviour
%   (See SPECtests1norm)
%---------------------------------------------------------------------------
      PxxC=PxxC.*sintr;
%---------------------------------------------------------------------------
%  Apply transfer function (includes conversion to accel and cal factor)
%---------------------------------------------------------------------------
      [mag,phase,fout,kend]=TF54000(nfftC/2);
      PxxC = PxxC .* (mag.*10^(-9)).^2;
      PxxC=PxxC(1:kend);
      fC=fC(1:kend);
%---------------------------------------------------------------------------
%  Smooth the spectra
%---------------------------------------------------------------------------
      NPD=64;
      NN=length(PxxC);
      [ff,PP]=smooth_spec(fC,PxxC,NPD);
      A=[ff,10*log10(PP)];
%---------------------------------------------------------------------------
%  f should be the same for all spectra; SAdb should be saved for plotting
%  in waterfall format
%--------------------------------------------------------------------------
      f=A(:,1);
      SAdb=A(:,2);
%--------------------------------------------------------------------------
%  Plot the data (just for single spectra)
%---------------------------------------------------------------------------
      if ichan==1
        fip = fopen(filev(ichunk,:),'w+');
%---------------------------------------------------------------------------
%        fprintf (fip,' %f \n',dbtime(ichunk,:));
%---------------------------------------------------------------------------
        fprintf (fip,' %f ',SAdb);
        fprintf (fip,'\n');
        fclose (fip);
%---------------------------------------------------------------------------
%  FPRINTF Write formatted data to file.
%   COUNT = FPRINTF(FID,FORMAT,A,...) formats the data in the real
%   part of matrix A (and in any additional matrix arguments), under
%   control of the specified FORMAT string, and writes it to the file
%   associated with file identifier FID.  COUNT is the number of bytes
%   successfully written. FID is an integer file identifier obtained
%   from FOPEN. It can also be 1 for standard output (the screen) or 2
%   for standard error. If FID is omitted, output goes to the screen.
%
%   FORMAT is a string containing C language conversion specifications.
%   Conversion specifications involve the character %, optional flags,
%   optional width and precision fields, optional subtype specifier, and
%   conversion characters d, i, o, u, x, X, f, e, E, g, G, c, and s.
%   See the Language Reference Guide or a C manual for complete details.
%
```

```
%   The special formats \n,\r,\t,\b,\f can be used to produce linefeed,
%   carriage return, tab, backspace, and formfeed characters respectively.
%   Use \\ to produce a backslash character and %% to produce the percent
%   character.
%---------------------------------------------------------------------------
        semilogx(f,SAdb,'b-')
        hold on
      elseif ichan==2
        semilogx(f,SAdb,'r-')
        fip = fopen(fileh(ichunk,:),'w+');
%---------------------------------------------------------------------------
%        fprintf (fip,' %f \n',dbtime(ichunk,:));
%---------------------------------------------------------------------------
       fprintf (fip,' %f ',SAdb);
        fprintf (fip,'\n');
        fclose (fip);
        hold on
      end
    end
  end %  of chunk loop
%---------------------------------------------------------------------------
%  HOLD   Hold current graph.
%   HOLD ON holds the current plot and all axis properties so that
%   subsequent graphing commands add to the existing graph.
%---------------------------------------------------------------------------
%  LOG    Natural logarithm.
%   LOG(X) is the natural logarithm of the elements of X.
%   Complex results are produced if X is not positive.
%---------------------------------------------------------------------------
%  Plot Legend
%---------------------------------------------------------------------------
%PLOT   Linear plot.
%   PLOT(X,Y) plots vector Y versus vector X. If X or Y is a matrix,
%   then the vector is plotted versus the rows or columns of the matrix,
%   whichever line up.  If X is a scalar and Y is a vector, length(Y)
%   disconnected points are plotted.
%
%   PLOT(Y) plots the columns of Y versus their index.
%   If Y is complex, PLOT(Y) is equivalent to PLOT(real(Y),imag(Y)).
%   In all other uses of PLOT, the imaginary part is ignored.
%
%   Various line types, plot symbols and colors may be obtained with
%   PLOT(X,Y,S) where S is a character string made from one element
%   from any or all the following 3 colunms:
%
%         y     yellow        .     point              -     solid
%         m     magenta       o     circle             :     dotted
%         c     cyan          x     x-mark             -.    dashdot
%         r     red           +     plus               --    dashed
%         g     green         *     star
%         b     blue          s     square
%         w     white         d     diamond
%         k     black         v     triangle (down)
%                             ^     triangle (up)
%                             <     triangle (left)
%                             >     triangle (right)
%                             p     pentagram
%                             h     hexagram
%
%   For example, PLOT(X,Y,'c+:') plots a cyan dotted line with a plus
%   at each data point; PLOT(X,Y,'bd') plots blue diamond at each data
```

```
%   point but does not draw any line.
%
%   PLOT(X1,Y1,S1,X2,Y2,S2,X3,Y3,S3,...) combines the plots defined by
%   the (X,Y,S) triples, where the X's and Y's are vectors or matrices
%   and the S's are strings.
%
%   For example, PLOT(X,Y,'y-',X,Y,'go') plots the data twice, with a
%   solid yellow line interpolating green circles at the data points.
%
%   The PLOT command, if no color is specified, makes automatic use of
%   the colors specified by the axes ColorOrder property.  The default
%   ColorOrder is listed in the table above for color systems where the
%   default is yellow for one line, and for multiple lines, to cycle
%   through the first six colors in the table.  For monochrome systems,
%   PLOT cycles over the axes LineStyleOrder property.
%
%   PLOT returns a column vector of handles to LINE objects, one
%   handle per line.
%
%   The X,Y pairs, or X,Y,S triples, can be followed by
%   parameter/value pairs to specify additional properties
%   of the lines.
%-------------------------------------------------------------------------
  shift=50;
  x=[1.5*10^(-3) 1.5*10^(-3) 0.8*10^(-1) 0.8*10^(-1)];
  y=[-156+shift -143+shift -143+shift -156+shift];
  patch(x,y,'w')
  text(2*10^(-3),-146+shift,'OSN-VERT     ');
  text(2*10^(-3),-152+shift,'OSN-HORI(rms) ');
  plot([0.3*10^(-1) 0.6*10^(-1)],[-146+shift -146+shift],'b-')
  plot([0.3*10^(-1) 0.6*10^(-1)],[-152+shift -152+shift],'r-')
%-------------------------------------------------------------------------
% draw the low and high Noise curves from Peterson
%-------------------------------------------------------------------------
  for m=1:2
    if m==1
      eval(['load ../matlab/nlnm_avd ;'])
      eval(['NM=nlnm_avd;']);
    elseif m==2
      eval(['load ../matlab/nhnm_avd ;'])
      eval(['NM=nhnm_avd;']);
    end
    km=1:length(NM);
    tm  =NM(km,1);        % periods(sec)
    fm  =NM(km,2);        % frequency(Hz)
    acc =NM(km,3);      % db refered to acceleration
    vel =NM(km,4);      % db refered to velocity
    dis =NM(km,5);      % db refered to displacement
    semilogx(fm,acc,'k--');
  end
  hold on
%-------------------------------------------------------------------------
% EVAL Execute string with MATLAB expression.
%   EVAL(s), where s is a string, causes MATLAB to execute
%   the string as an expression or statement.
%   [X,Y,Z,...] = EVAL(s) returns output arguments from the
%   expression in string s.
%
%   The input strings to EVAL are often created by
%   concatenating substrings and variables inside square
%   brackets. For example:
```

```
%
%   Generate a sequence of matrices named M1 through M12:
%
%       for n = 1:12
%          eval(['M' num2str(n) ' = magic(n)'])
%       end
%
%   Run a selected M-file script.  The strings making up
%   the rows of matrix D must all have the same length.
%
%       D = ['odedemo '
%            'quaddemo'
%            'fitdemo '];
%       n = input('Select a demo number: ');
%       eval(D(n,:))
%---------------------------------------------------------------------------
%  End of Script
```

- **t06_timeplot.m**

```
clf
clear
  doit=0;

  fileh(1,:)  = '../spectra/test01H.spec';
      ect.

nfftC=32768;
  [mag,phase,fout,kend]=TF54000(nfftC/2);
  for ichan=1:2
    if ichan==1
      orientation='z';

      fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BHZ');
      C1=fread(fid,131072,'float');
      fclose(fid);

    elseif ichan==2
      orientation='n';

      fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH1');
      C2=fread(fid,131072,'float');
      fclose(fid);

      fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH2');
      C3=fread(fid,131072,'float');
      fclose(fid);

      C1=sqrt(C2.^2+C3.^2);

    end

    kmm=length(C1);
    kmm=kmm/131072;
    kmm=fix(kmm);

    for ichunk=1:kmm
      ist=(ichunk-1)*131072+1;
      C2=C1(ist:1500:ist+131072-1);
      FC=20;
```

```
      nfftC=32768;
      windowC=hanning(nfftC);
      noverlapC=0.75*nfftC;
      dflagC='linear';
      plot(C2);
    end
  end
```

- **t07_threechan.m**

```
clf
clear
doit=0;
%
  fileh(1,:)  = '../spectra/test01H.spec';
      ect.

  nfftC=32768;
  [mag,phase,fout,kend]=TF54000(nfftC/2);
%
  for ichan=1:3
    if ichan==1
      orientation='z';
      fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BHZ');
      C1=fread(fid,131072,'float');
      fclose(fid);
    elseif ichan==2
      orientation='n';
      fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH1');
      C2=fread(fid,131072,'float');
      fclose(fid);
      C1=C2;
    elseif ichan==3
      orientation='n1';
      fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH2');
      C3=fread(fid,131072,'float');
      fclose(fid);
      C1=C3;
%
%
    end

    kmm=length(C1);
    kmm=kmm/131072;
    kmm=fix(kmm);
    for ichunk=1:kmm
      ist=(ichunk-1)*131072+1;
      C2=C1(ist:ist+131072-1);
      FC=20;
      nfftC=32768;
      windowC=hanning(nfftC);
      noverlapC=0.75*nfftC;
      dflagC='linear';
      [PxxC,fC]=psd(C2,nfftC,FC,windowC,noverlapC,dflagC);
      NN=length(fC);
      PxxC=PxxC(2:NN-1);
      fC=fC(2:NN-1);
      sintr=0.05;
      NN=length(PxxC);
```

```
      PxxC=PxxC.*sintr;
      [mag,phase,fout,kend]=TF54000(nfftC/2);
      PxxC = PxxC .* (mag.*10^(-9)).^2;
      PxxC=PxxC(1:kend);
      fC=fC(1:kend);
      NPD=64;
      NN=length(PxxC);
      [ff,PP]=smooth_spec(fC,PxxC,NPD);
      A=[ff,10*log10(PP)];
      f=A(:,1);
      SAdb=A(:,2);
      if ichan==1
        fip = fopen(filev(ichunk,:),'w+');
        fprintf (fip,' %f ',SAdb);
        fprintf (fip,'\n');
        fclose (fip);
        semilogx(f,SAdb,'b-')
        axis([10^-1 10^1 -165 -95]);
        hold on
      elseif ichan==2
        semilogx(f,SAdb,'r--')
        axis([10^-1 10^1 -165 -95]);
        fip = fopen(fileh(ichunk,:),'w+');
        fprintf (fip,' %f ',SAdb);
        fprintf (fip,'\n');
        fclose (fip);
        hold on
      elseif ichan==3
            semilogx(f,SAdb,'c:')
            axis([10^-1 10^1 -165 -95]);
        fip = fopen(fileh1(ichunk,:),'w+');
        fprintf (fip,' %f ',SAdb);
        fprintf (fip,'\n');
        fclose (fip);
        hold on
      end
    end
  end %  of chunk loop
  shift=50;
  x=[1.5*10^(-3) 1.5*10^(-3) 0.8*10^(-1) 0.8*10^(-1)];
  y=[-162+shift -143+shift -143+shift -162+shift];
  patch(x,y,'w')
  text(2*10^(-3),-146+shift,'OSN-VERT     ');
  text(2*10^(-3),-152+shift,'OSN-HORI-1     ');
  text(2*10^(-3),-158+shift,'OSN-HORI-2     ');
  plot([0.3*10^(-1) 0.6*10^(-1)],[-146+shift -146+shift],'b-')
  axis([10^-1 10^1 -165 -95]);
  plot([0.3*10^(-1) 0.6*10^(-1)],[-152+shift -152+shift],'r-')
  axis([10^-1 10^1 -165 -95]);
  plot([0.3*10^(-1) 0.6*10^(-1)],[-158+shift -158+shift],'c-')
  axis([10^-1 10^1 -165 -95]);
  for m=1:2
    if m==1
      eval(['load ../matlab/nlnm_avd ;'])
      eval(['NM=nlnm_avd;']);
    elseif m==2
      eval(['load ../matlab/nhnm_avd ;'])
      eval(['NM=nhnm_avd;']);
    end
    km=1:length(NM);
    tm  =NM(km,1);        % periods(sec)
```

```
    fm  =NM(km,2);        % frequency(Hz)
    acc =NM(km,3);      % db refered to acceleration
    vel =NM(km,4);      % db refered to velocity
    dis =NM(km,5);      % db refered to displacement
    semilogx(fm,acc,'k--');
    axis([10^-1 10^1 -165 -95]);
  end
  hold on
```

- **t08s_threechan.m**

```
clf
clear
%
doit=0;
%
  fileh(1,:)  = '../spectra/test01H.spec';
      ect.


  nfftC=32768;
  [mag,phase,fout,kend]=TF54000(nfftC/2);
%
  for ichan=1:3
    if ichan==1
      orientation='z';
%-------------------------------------------------------------------
%
%  The data files for day 62 have already been uncompressed
%-------------------------------------------------------------------
      fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BHZ');
      C1=fread(fid,131072,'float');
      fclose(fid);
    elseif ichan==2
      orientation='n';
%
      fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH1');
      C2=fread(fid,131072,'float');
      fclose(fid);
      C1=C2;
%
    elseif ichan==3
      orientation='n1';

      fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH2');
      C3=fread(fid,131072,'float');
      fclose(fid);
      C1=C3;
%
%
    end

    kmm=length(C1);
    kmm=kmm/131072;
    kmm=fix(kmm);

    for ichunk=1:kmm
      ist=(ichunk-1)*131072+1;
      C2=C1(ist:ist+131072-1);
```

```
      FC=20;
      nfftC=32768;
      windowC=hanning(nfftC);
      noverlapC=0.75*nfftC;
      dflagC='linear';
      [PxxC,fC]=psd(C2,nfftC,FC,windowC,noverlapC,dflagC);
      NN=length(fC);
      PxxC=PxxC(2:NN-1);
      fC=fC(2:NN-1);
%---------------------------------------------------------------------------
%  Amplitude adjustments
%---------------------------------------------------------------------------
      sintr=0.05;
      NN=length(PxxC);
%---------------------------------------------------------------------------
%  Multiply by delt to get correct root(Hz) behaviour
%   (See SPECtests1norm)
%--------------------------------------------------------------------------
      PxxC=PxxC.*sintr;
%--------------------------------------------------------------------------
%  Apply transfer function (includes conversion to accel and cal factor)
%---------------------------------------------------------------------------
      [mag,phase,fout,kend]=TF54000(nfftC/2);
      PxxC = PxxC .* (mag.*10^(-9)).^2;
      PxxC=PxxC(1:kend);
      fC=fC(1:kend);
%---------------------------------------------------------------------------
%  Smooth the spectra
%---------------------------------------------------------------------------
      NPD=64;
      NN=length(PxxC);
      [ff,PP]=smooth_spec(fC,PxxC,NPD);
      A=[ff,10*log10(PP)];
%---------------------------------------------------------------------------
%  f should be the same for all spectra; SAdb should be saved for plotting
%  in waterfall format
%---------------------------------------------------------------------------
      f=A(:,1);
      SAdb=A(:,2);
%---------------------------------------------------------------------------
%  Plot the data (just for single spectra)
%---------------------------------------------------------------------------
      if ichan==1
        fip = fopen(filev(ichunk,:),'w+');
%---------------------------------------------------------------------------
%        fprintf (fip,' %f \n',dbtime(ichunk,:));
%---------------------------------------------------------------------------
        fprintf (fip,' %f ',SAdb);
        fprintf (fip,'\n');
        fclose (fip);

        semilogx(f,SAdb,'b-')
        hold on
      elseif ichan==2
        semilogx(f,SAdb,'r-')
        fip = fopen(fileh(ichunk,:),'w+');
%---------------------------------------------------------------------------
%        fprintf (fip,' %f \n',dbtime(ichunk,:));
%---------------------------------------------------------------------------
        fprintf (fip,' %f ',SAdb);
        fprintf (fip,'\n');
```

```
        fclose (fip);
        hold on
      elseif ichan==3
            semilogx(f,SAdb,'c-')
        fip = fopen(fileh1(ichunk,:),'w+');
        fprintf (fip,' %f ',SAdb);
        fprintf (fip,'\n');
        fclose (fip);
        hold on
      end
    end
  end %  of chunk loop

  shift=50;
  x=[1.5*10^(-3) 1.5*10^(-3) 0.8*10^(-1) 0.8*10^(-1)];
  y=[-162+shift -143+shift -143+shift -162+shift];
  patch(x,y,'w')
  text(2*10^(-3),-146+shift,'OSN-VERT     ');
  text(2*10^(-3),-152+shift,'OSN-HORI-1     ');
  text(2*10^(-3),-158+shift,'OSN-HORI-2     ');
  plot([0.3*10^(-1) 0.6*10^(-1)],[-146+shift -146+shift],'b-')
  plot([0.3*10^(-1) 0.6*10^(-1)],[-152+shift -152+shift],'r-')
  plot([0.3*10^(-1) 0.6*10^(-1)],[-158+shift -158+shift],'c-')
%---------------------------------------------------------------------------
% draw the low and high Noise curves from Peterson
%---------------------------------------------------------------------------
  for m=1:2
    if m==1
      eval(['load ../matlab/nlnm_avd ;'])
      eval(['NM=nlnm_avd;']);
    elseif m==2
      eval(['load ../matlab/nhnm_avd ;'])
      eval(['NM=nhnm_avd;']);
    end
    km=1:length(NM);
    tm  =NM(km,1);        % periods(sec)
    fm  =NM(km,2);        % frequency(Hz)
    acc =NM(km,3);      % db refered to acceleration
    vel =NM(km,4);      % db refered to velocity
    dis =NM(km,5);      % db refered to displacement
    semilogx(fm,acc,'k--');
  end
  hold on
%
%  End of Script
```

- **t09_OSNcohere.m**

```
clf
clear
  doit=0;

  fileH1H2(1,:)  = '../spectra/test01H.spec';
      ect.


  nfftC=32768;

  fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BHZ');
```

```
Z=fread(fid,131072,'float');
fclose(fid);

fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH1');
H1=fread(fid,131072,'float');
fclose(fid);

fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH2');
H2=fread(fid,131072,'float');
fclose(fid);


kmm=length(Z);
kmm=kmm/(131072);
kmm=fix(kmm);

for ichunk=1:kmm
  ist=(ichunk-1)*131072+1;
  Z=Z(ist:ist+131072-1);
  H1=H1(ist:ist+131072-1);
  H2=H2(ist:ist+131072-1);
  FC=20;
  nfftC=32768;
  windowC=hanning(nfftC);
  noverlapC=0.75*nfftC;
  dflagC='linear';

  for ichan=1:4
    if ichan==1

      [PZH1,fC]=cohere(Z,H1,nfftC,FC,windowC,noverlapC,dflagC);
      PZH1=PZH1(2:length(PZH1));
      fC=fC(2:length(PZH1));
      NPD=64;
      [ffZH1,PPZH1]=smooth_spec(fC,PZH1,NPD);
      fip = fopen(fileZH1(ichunk,:),'w+');
      fprintf (fip,' %f ',PPZH1);
      fprintf (fip,'\n');
      fclose (fip);
      semilogx(ffZH1,PPZH1,'-');
      ylabel('coherence coefficient');
      xlabel('frequency (Hz)');
      title('Coherece between OSN-Z and OSN-H1');

    elseif ichan==2

      [PH1H2,fC]=cohere(H1,H2,nfftC,FC,windowC,noverlapC,dflagC);
      PH1H2=PH1H2(2:length(PH1H2));
      fC=fC(2:length(PH1H2));
      NPD=64;
      [ffH1H2,PPH1H2]=smooth_spec(fC,PH1H2,NPD);
      fip = fopen(fileH1H2(ichunk,:),'w+');
      fprintf (fip,' %f ',PPH1H2);
      fprintf (fip,'\n');
      fclose (fip);
      semilogx(ffH1H2,PPH1H2,'-');
      ylabel('coherence coefficient');
      xlabel('frequency (Hz)');
      title('Coherece between OSN-H1 and OSN-H2');

    elseif ichan==3
```

```
        [PZH2,fC]=cohere(Z,H2,nfftC,FC,windowC,noverlapC,dflagC);
        PZH2=PZH2(2:length(PZH2));
        fC=fC(2:length(PZH2));
        NPD=64;
        [ffZH2,PPZH2]=smooth_spec(fC,PZH2,NPD);
        fip = fopen(fileZH2(ichunk,:),'w+');
        fprintf (fip,' %f ',PPZH2);
        fprintf (fip,'\n');
        fclose (fip);
        semilogx(ffZH2,PPZH2,'-');
        ylabel('coherence coefficient');
        xlabel('frequency (Hz)');
        title('Coherece between OSN-Z and OSN-H2');

      elseif ichan==4

        subplot(3,1,1), semilogx(ffZH1,PPZH1,'-');
        ylabel('coherence coefficient');
        xlabel('frequency (Hz)');
        title('Coherece between OSN-Z and OSN-H1');
        subplot(3,1,2), semilogx(ffH1H2,PPH1H2,'-');
        ylabel('coherence coefficient');
        xlabel('frequency (Hz)');
        title('Coherece between OSN-H1 and OSN-H2');
        subplot(3,1,3), semilogx(ffZH2,PPZH2,'-');
        ylabel('coherence coefficient');
        xlabel('frequency (Hz)');
        title('Coherece between OSN-Z and OSN-H2');

      end

      fip = fopen('fC.dat','w+');
      fprintf (fip,' %f ',fC);
      fprintf (fip,'\n');
      fclose (fip);
    end
  end
```

- **t10_H2Ocsd.m**

```
clf
%clear

  nfftC=32768;

%  load /osn1/b3s2/OSN/H2O/work/h20_121_HH

  kmm=length(C1);
  kmm=kmm/(131072);
  kmm=fix(kmm)

  for ichunk=1:kmm
    ronde=ichunk
    ist=(ichunk-1)*131072+1;
    Z=C1(ist:ist+131072-1);
    H1=C2(ist:ist+131072-1);
    H2=C3(ist:ist+131072-1);
    FC=20;
    nfftC=32768;
```

```
    windowC=hanning(nfftC);
    noverlapC=0.75*nfftC;
    dflagC='linear';

    for ichan=1:4
      if ichan==1

        [CZH1,fC]=csd(Z,H1,nfftC,FC,windowC,noverlapC,dflagC);
        CZH1mag=10*log10(abs(CZH1));

      elseif ichan==2

        [CH1H2,fC]=csd(H1,H2,nfftC,FC,windowC,noverlapC,dflagC);
        CH1H2mag=10*log10(abs(CH1H2));

      elseif ichan==3

        [CZH2,fC]=csd(Z,H2,nfftC,FC,windowC,noverlapC,dflagC);
        CZH2mag=10*log10(abs(CZH2));

      elseif ichan==4

        subplot(3,1,1), semilogx(fC,angle(CZH1),'-');
        ylabel('phase/angle (rad) ');
        TITLE=sprintf('Phase/angle plots H2O-Z and H2O-H1       CHUNK:
%d\n',ichunk);
        title(TITLE);

        subplot(3,1,2), semilogx(fC,angle(CH1H2),'-');
        ylabel('phase/angle (rad) ');
        title('Phase/angle plots H2O-H1 and H2O-H2');

        subplot(3,1,3), semilogx(fC,angle(CZH2),'-');
        ylabel('phase/angle (rad)');
        xlabel('frequency (Hz)');
        title('Phase/angle plots H2O-Z and H2O-H2');
        orient landscape
        print
        end
      end
    end
```

- **t10_OSNcsd.m**

```
clf
%clear

  nfftC=32768;

  fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BHZ');
  Zi=fread(fid,131072,'float');
  fclose(fid);

  fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH1');
  H1i=fread(fid,131072,'float');
  fclose(fid);

  fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH2');
  H2i=fread(fid,131072,'float');
```

```
fclose(fid);

kmm=length(C1);
kmm=kmm/(131072);
kmm=fix(kmm)

for ichunk=1:kmm
  ist=(ichunk-1)*131072+1;
  Z=Zi(ist:ist+131072-1);
  H1=H1i(ist:ist+131072-1);
  H2=H2i(ist:ist+131072-1);
  FC=20;
  nfftC=32768;
  windowC=hanning(nfftC);
  noverlapC=0.75*nfftC;
  dflagC='linear';

  for ichan=1:4
    if ichan==1

      [CZH1,fC]=csd(Z,H1,nfftC,FC,windowC,noverlapC,dflagC);
      CZH1mag=10*log10(abs(CZH1));

    elseif ichan==2

      [CH1H2,fC]=csd(H1,H2,nfftC,FC,windowC,noverlapC,dflagC);
      CH1H2mag=10*log10(abs(CH1H2));

    elseif ichan==3

      [CZH2,fC]=csd(Z,H2,nfftC,FC,windowC,noverlapC,dflagC);
      CZH2mag=10*log10(abs(CZH2));

    elseif ichan==4

      subplot(3,1,1), semilogx(fC,angle(CZH1),'-');
      ylabel('phase/angle (rad) ');
      TITLE=sprintf('Phase/angle plots OSN-Z and OSN-H1       CHUNK:
    % d\n',ichunk);
      title(TITLE);

      subplot(3,1,2), semilogx(fC,angle(CH1H2),'-');
      ylabel('phase/angle (rad) ');
      title('Phase/angle plots OSN-H1 and OSN-H2');

      subplot(3,1,3), semilogx(fC,angle(CZH2),'-');
      ylabel('phase/angle (rad)');
      xlabel('frequency (Hz)');
      title('Phase/angle plots OSN-Z and OSN-H2');

      orient landscape
      print
      end
    end
  end
```

- **t11_OSNpsd+coh.m**

```
clf
```

```
clear
doit=0;

nfftC=32768;

fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BHZ');
Zi=fread(fid,'float');
fclose(fid);

fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH1');
H1i=fread(fid,'float');
fclose(fid);

fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH2');
H2i=fread(fid,'float');
fclose(fid);


kmm=length(Zi);
kmm=kmm/(131072);
kmm=fix(kmm)

for ichunk=1:kmm
  ist=(ichunk-1)*131072+1;
  Z=Zi(ist:ist+131072-1);
  H1=H1i(ist:ist+131072-1);
  H2=H2i(ist:ist+131072-1);
  FC=20;
  nfftC=32768;
  windowC=hanning(nfftC);
  noverlapC=0.75*nfftC;
  dflagC='linear';

  for ichan=1:3
    if ichan==1

    [PZH1,fC]=cohere(Z,H1,nfftC,FC,windowC,noverlapC,dflagC);
    PZH1=PZH1(2:length(PZH1));
    fC=fC(2:length(PZH1));
    NPD=64;
    [ffZH1,PPZH1]=smooth_spec(fC,PZH1,NPD);

    C2=Z;
    cloop=1

    elseif ichan==2

    [PH1H2,fC]=cohere(H1,H2,nfftC,FC,windowC,noverlapC,dflagC);
    PH1H2=PH1H2(2:length(PH1H2));
    fC=fC(2:length(PH1H2));
    NPD=64;
    [ffH1H2,PPH1H2]=smooth_spec(fC,PH1H2,NPD);

    C2=H1;
    cloop=2

    elseif ichan==3

    [PZH2,fC]=cohere(Z,H2,nfftC,FC,windowC,noverlapC,dflagC);
    PZH2=PZH2(2:length(PZH2));
    fC=fC(2:length(PZH2));
```

```
      NPD=64;
      [ffZH2,PPZH2]=smooth_spec(fC,PZH2,NPD);

      C2=H2;
      cloop=3
      end

    [PxxC,fC]=psd(C2,nfftC,FC,windowC,noverlapC,dflagC);
    NN=length(fC);
    PxxC=PxxC(2:NN-1);
    fC=fC(2:NN-1);
    sintr=0.05;
    NN=length(PxxC);
    PxxC=PxxC.*sintr;
    [mag,phase,fout,kend]=TF54000(nfftC/2);
    PxxC = PxxC .* (mag.*10^(-9)).^2;
    PxxC=PxxC(1:kend);
    fC=fC(1:kend);
    NPD=64;
    NN=length(PxxC);
    [ff,PP]=smooth_spec(fC,PxxC,NPD);
    A=[ff,10*log10(PP)];
    f=A(:,1);
    SAdb=A(:,2);


      if ichan==1
      SAdbZ=SAdb;
      fZ=f;
      ploop=1
      elseif ichan==2
      SAdbH1=SAdb;
      fH1=f;
      ploop=2
      elseif ichan==3
      SAdbH2=SAdb;
      fH2=f;
      ploop=3
      end

    end

  subplot('position',[0.1500 0.7475 0.8000 0.1725 ])
  semilogx(ffZH1,PPZH1,'-');
    if ichunk==1
    title('Coherece between OSN-Z and OSN-H1          CHUNK 1');
    elseif ichunk==2
    title('Coherece between OSN-Z and OSN-H1          CHUNK 2');
    elseif ichunk==3
    title('Coherece between OSN-Z and OSN-H1          CHUNK 3');
    elseif ichunk==4
    title('Coherece between OSN-Z and OSN-H1          CHUNK 4');
    elseif ichunk==5
    title('Coherece between OSN-Z and OSN-H1          CHUNK 5');
    elseif ichunk==6
    title('Coherece between OSN-Z and OSN-H1          CHUNK 6');
    elseif ichunk==7
    title('Coherece between OSN-Z and OSN-H1          CHUNK 7');
    elseif ichunk==8
    title('Coherece between OSN-Z and OSN-H1          CHUNK 8');
    elseif ichunk==9
```

```
    title('Coherece between OSN-Z and OSN-H1          CHUNK 9');
    elseif ichunk==10
    title('Coherece between OSN-Z and OSN-H1          CHUNK 10');
    elseif ichunk==11
    title('Coherece between OSN-Z and OSN-H1          CHUNK 11');
    elseif ichunk==12
    title('Coherece between OSN-Z and OSN-H1          CHUNK 12');
    elseif ichunk==13
    title('Coherece between OSN-Z and OSN-H1          CHUNK 13');
    end
  axis([10^-4 10^1 0 1]);
  ylabel('coherence coefficient');

  subplot('position',[0.1500 0.525 0.8000 0.1725  ])
  semilogx(ffH1H2,PPH1H2,'-');
  axis([10^-4 10^1 0 1]);
  ylabel('coherence coefficient');
  title('Coherece between OSN-H1 and OSN-H2                   ');

  subplot('position',[0.1500 0.3025 0.8000 0.1725 ])
  semilogx(ffZH2,PPZH2,'-');
  ylabel('coherence coefficient');
  title('Coherece between OSN-Z and OSN-H2                    ');

  subplot('position',[0.1500 0.08 0.8000 0.1725 ])
  semilogx (fZ,SAdbZ,'-')
  hold on
  semilogx(fH1,SAdbH1,'-.')
  hold on
  semilogx(fH2,SAdbH2,':')
  xlabel('frequency (Hz)');
  ylabel('psd');
  title('Power Spectral Density                               ');
  hold on
  shift=50;
  x=[1.2*10^(-3) 1.2*10^(-3) 0.8*10^(-1) 0.8*10^(-1)];
  y=[-163+shift -121+shift -121+shift -163+shift];
  patch(x,y,'w')
  text(2*10^(-3),-126+shift,'OSN-Z        ');
  text(2*10^(-3),-142+shift,'OSN-H1   ');
  text(2*10^(-3),-157+shift,'OSN-H2   ');
  plot([0.3*10^(-1) 0.6*10^(-1)],[-126+shift -126+shift],'-')
  plot([0.3*10^(-1) 0.6*10^(-1)],[-142+shift -142+shift],'-.')
  plot([0.3*10^(-1) 0.6*10^(-1)],[-157+shift -157+shift],':')

  for m=1:2
    if m==1
    eval(['load ../matlab/nlnm_avd ;'])
    eval(['NM=nlnm_avd;']);
    elseif m==2
    eval(['load ../matlab/nhnm_avd ;'])
    eval(['NM=nhnm_avd;']);
    end
  km=1:length(NM);
  tm  =NM(km,1);        % periods(sec)
  fm  =NM(km,2);        % frequency(Hz)
  acc =NM(km,3);  % db refered to acceleration
  vel =NM(km,4);  % db refered to velocity
  dis =NM(km,5);  % db refered to displacement
  semilogx(fm,acc,'k--')
  axis([10^-4 10^1 -200 -40]);
```

```
  end
  hold off
  set (gcf,'paperposition',[0 0 8.5 10.8],'paperorientation','portrait')
  print
  end
```

- **t12_H2Opsd+coh.m**

```
  clf
% clear
  doit=0;

  nfftC=32768;

%  load /osn1/b3s2/OSN/H2O/work/h20_121_HH

%  kmm=length(C1);
%  kmm=kmm/(131072);
%  kmm=fix(kmm)
%  kmm=1


% for ichunk=1:kmm
  for ichunk=13
    ronde=ichunk
    ist=(ichunk-1)*131072+1;
    Z=C1(ist:ist+131072-1);
    H1=C2(ist:ist+131072-1);
    H2=C3(ist:ist+131072-1);
    FC=20;
    nfftC=32768;
    windowC=hanning(nfftC);
    noverlapC=0.75*nfftC;
    dflagC='linear';

    for ichan=1:3
      if ichan==1

      [PZH1,fC]=cohere(Z,H1,nfftC,FC,windowC,noverlapC,dflagC);
      PZH1=PZH1(2:length(PZH1));
      fC=fC(2:length(PZH1));
      NPD=64;
      [ffZH1,PPZH1]=smooth_spec(fC,PZH1,NPD);

      P2=Z;
      cloop=1

      elseif ichan==2

      [PH1H2,fC]=cohere(H1,H2,nfftC,FC,windowC,noverlapC,dflagC);
      PH1H2=PH1H2(2:length(PH1H2));
      fC=fC(2:length(PH1H2));
      NPD=64;
      [ffH1H2,PPH1H2]=smooth_spec(fC,PH1H2,NPD);

      P2=H1;
      cloop=2

      elseif ichan==3
```

```
      [PZH2,fC]=cohere(Z,H2,nfftC,FC,windowC,noverlapC,dflagC);
      PZH2=PZH2(2:length(PZH2));
      fC=fC(2:length(PZH2));
      NPD=64;
      [ffZH2,PPZH2]=smooth_spec(fC,PZH2,NPD);

      P2=H2;
      cloop=3
      end

    [PxxC,fC]=psd(P2,nfftC,FC,windowC,noverlapC,dflagC);
    NN=length(fC);
    PxxC=PxxC(2:NN-1);
    fC=fC(2:NN-1);
    sintr=0.05;
    NN=length(PxxC);
    PxxC=PxxC.*sintr;
    [mag,phase,fout,kend]=TF54000(nfftC/2);
    PxxC = PxxC .* (mag.*10^(-9)).^2;
    PxxC=PxxC(1:kend);
    fC=fC(1:kend);
    NPD=64;
    NN=length(PxxC);
    [ff,PP]=smooth_spec(fC,PxxC,NPD);
    A=[ff,10*log10(PP)];
    f=A(:,1);
    SAdb=A(:,2);


      if ichan==1
      SAdbZ=SAdb;
      fZ=f;
      ploop=1
      elseif ichan==2
      SAdbH1=SAdb;
      fH1=f;
      ploop=2
      elseif ichan==3
      SAdbH2=SAdb;
      fH2=f;
      ploop=3
      end

    end

  subplot('position',[0.1500 0.7475 0.8000 0.1725 ])
  semilogx(ffZH1,PPZH1,'-');
    if ichunk==1
    title('Coherece between H2O-Z and H2O-H1          CHUNK 1');
    elseif ichunk==2
    title('Coherece between H2O-Z and H2O-H1          CHUNK 2');
    elseif ichunk==3
    title('Coherece between H2O-Z and H2O-H1          CHUNK 3');
    elseif ichunk==4
    title('Coherece between H2O-Z and H2O-H1          CHUNK 4');
    elseif ichunk==5
    title('Coherece between H2O-Z and H2O-H1          CHUNK 5');
    elseif ichunk==6
    title('Coherece between H2O-Z and H2O-H1          CHUNK 6');
    elseif ichunk==7
    title('Coherece between H2O-Z and H2O-H1          CHUNK 7');
```

```
    elseif ichunk==8
    title('Coherece between H2O-Z and H2O-H1          CHUNK 8');
    elseif ichunk==9
    title('Coherece between H2O-Z and H2O-H1          CHUNK 9');
    elseif ichunk==10
    title('Coherece between H2O-Z and H2O-H1          CHUNK 10');
    elseif ichunk==11
    title('Coherece between H2O-Z and H2O-H1          CHUNK 11');
    elseif ichunk==12
    title('Coherece between H2O-Z and H2O-H1          CHUNK 12');
    elseif ichunk==13
    title('Coherece between H2O-Z and H2O-H1          CHUNK 13');
    end
  axis([10^-4 10^1 0 1]);
  ylabel('coherence coefficient');

  subplot('position',[0.1500 0.525 0.8000 0.1725  ])
  semilogx(ffH1H2,PPH1H2,'-');
  axis([10^-4 10^1 0 1]);
  ylabel('coherence coefficient');
  title('Coherence between H2O-H1 and H2O-H2                   ');

  subplot('position',[0.1500 0.3025 0.8000 0.1725 ])
  semilogx(ffZH2,PPZH2,'-');
  axis([10^-4 10^1 0 1]);
  ylabel('coherence coefficient');
  title('Coherence between H2O-Z and H2O-H2                    ');

  subplot('position',[0.1500 0.08 0.8000 0.1725 ])
  semilogx (fZ,SAdbZ,'-')
  hold on
  semilogx(fH1,SAdbH1,'-.')
  hold on
  semilogx(fH2,SAdbH2,':')
  xlabel('frequency (Hz)');
  ylabel('psd');
  title('Power Spectral Density H2O                              ');
  hold on
  shift=50;
  x=[1.2*10^(-3) 1.2*10^(-3) 0.8*10^(-1) 0.8*10^(-1)];
  y=[-163+shift -121+shift -121+shift -163+shift];
  patch(x,y,'w')
  text(2*10^(-3),-126+shift,'H2O-Z        ');
  text(2*10^(-3),-142+shift,'H2O-H1   ');
  text(2*10^(-3),-157+shift,'H2O-H2   ');
  plot([0.3*10^(-1) 0.6*10^(-1)],[-126+shift -126+shift],'-')
  plot([0.3*10^(-1) 0.6*10^(-1)],[-142+shift -142+shift],'-.')
  plot([0.3*10^(-1) 0.6*10^(-1)],[-157+shift -157+shift],':')

  for m=1:2
    if m==1
    eval(['load /u/zeldenru/OSN/matlab/nlnm_avd ;'])
    eval(['NM=nlnm_avd;']);
    elseif m==2
    eval(['load /u/zeldenru/OSN/matlab/nhnm_avd ;'])
    eval(['NM=nhnm_avd;']);
    end
  km=1:length(NM);
  tm  =NM(km,1);        % periods(sec)
  fm  =NM(km,2);        % frequency(Hz)
  acc =NM(km,3);  % db refered to acceleration
```

```
  vel =NM(km,4);  % db refered to velocity
  dis =NM(km,5);  % db refered to displacement
  semilogx(fm,acc,'k--')
  axis([10^-4 10^1 -200 -40]);
  end
  hold off
  set (gcf,'paperposition',[0 0 8.5 10.8],'paperorientation','portrait')
  print
  end
```

- **t13_OSNtransf.m**

```
clf
clear
  doit=0;

  nfftC=32768;

  fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BHZ');
  Zi=fread(fid,'float');
  fclose(fid);

  fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH1');
  H1i=fread(fid,'float');
  fclose(fid);

  fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH2');
  H2i=fread(fid,'float');
  fclose(fid);


  kmm=length(Zi);
  kmm=kmm/(131072);
  kmm=fix(kmm)

  for ichunk=1:kmm
    ist=(ichunk-1)*131072+1;
    Z=Zi(ist:ist+131072-1);
    H1=H1i(ist:ist+131072-1);
    H2=H2i(ist:ist+131072-1);
    FC=20;
    nfftC=32768;
    windowC=hanning(nfftC);
    noverlapC=0.75*nfftC;
    dflagC='linear';

    for ichan=1:3
      if ichan==1

      [PZH1t,fC]=cohere(Z,H1,nfftC,FC,windowC,noverlapC,dflagC);
      PZH1=(PZH1t).^0.5;
 %     length(PZH1)=length(PZH1t);
 %     PZH1=PZH1(2:length(PZH1));
 %     fC=fC(2:length(PZH1));
 %     NPD=64;
 %     [ffZH1,PPZH1]=smooth_spec(fC,PZH1,NPD);

      C2=Z;
      cloop=1
```

```
      elseif ichan==2

      [PH1H2t,fC]=cohere(H1,H2,nfftC,FC,windowC,noverlapC,dflagC);
      PH1H2=(PH1H2t).^0.5;
%      PH1H2=PH1H2(2:length(PH1H2));
%      fC=fC(2:length(PH1H2));
%      NPD=64;
%      [ffH1H2,PPH1H2]=smooth_spec(fC,PH1H2,NPD);

      C2=H1;
      cloop=2

      elseif ichan==3

      [PZH2t,fC]=cohere(Z,H2,nfftC,FC,windowC,noverlapC,dflagC);
      PZH2=(PZH2t).^0.5;
%      PZH2=PZH2(2:length(PZH2));
%      fC=fC(2:length(PZH2));
%      NPD=64;
%      [ffZH2,PPZH2]=smooth_spec(fC,PZH2,NPD);

      C2=H2;
      cloop=3
      end

    [PxxCt,fC]=psd(C2,nfftC,FC,windowC,noverlapC,dflagC);
%    NN=length(fC);
%    PxxC=PxxCt(2:NN-1);
%    fC=fC(2:NN-1);
%    sintr=0.05;
%    NN=length(PxxC);
%    PxxC=PxxC.*sintr;
%    [mag,phase,fout,kend]=TF54000(nfftC/2);
%    PxxC = PxxC .* (mag.*10^(-9)).^2;
%    PxxC=PxxC(1:kend);
%    fC=fC(1:kend);
%    NPD=64;
%    NN=length(PxxC);
%    [ff,PP]=smooth_spec(fC,PxxC,NPD);
%    A=[ff,10*log10(PP)];
%    f=A(:,1);
%    SAdb=A(:,2);


      if ichan==1
      PxxCZ=PxxCt;
%      fZ=f;
      ploop=1
      elseif ichan==2
      PxxCH1=PxxCt;
%      fH1=f;
      ploop=2
      elseif ichan==3
      PxxCH2=PxxCt;
%      fH2=f;
      ploop=3
      end

    end

  transZH1=(PZH1).*((PxxCZ)./(PxxCH1)).^0.5;
```

```
  transH2H1=(PH1H2).*((PxxCH2)./(PxxCH1)).^0.5;
  transZH2=(PZH2).*((PxxCZ)./(PxxCH2)).^0.5;

  subplot(3,1,1), semilogx(fC,20.*log10(abs(transZH1)),'-');
  axis([10^-4 10^0 -75 25]);
  ylabel('magnitude (dB) ');
  TITLE=sprintf('Transfer Function plots OSN-Z and OSN-H1       CHUNK:
%d\n',ichunk);
  title(TITLE);

  subplot(3,1,2), semilogx(fC,20.*log10(abs(transH2H1)),'-');
  axis([10^-4 10^0 -75 25]);
  ylabel('magnitude (dB) ');
  title('Transfer Function plots OSN-H1 and OSN-H2');

  subplot(3,1,3), semilogx(fC,20.*log10(abs(transZH2)),'-');
  axis([10^-4 10^0 -75 25]);
  ylabel('magnitude (dB)');
  xlabel('frequency (Hz)');
  title('Transfer Function plots OSN-Z and OSN-H2');
  orient landscape

  print
  end
```

- **t14s_H2Otransf.m**

```
clf
clear

  fileH2H1(1,:)  = '../spectra/H2Otest01H1H2.spec';
      ect.


%  load /osn1/b3s2/OSN/H2O/work/h20_121_HH

  kmm=length(C1);
  kmm=kmm/(131072);
  kmm=fix(kmm)

  for ichunk=1
    ronde=ichunk
    ist=(ichunk-1)*131072+1;
    Z=C1(ist:ist+131072-1);
    H1=C2(ist:ist+131072-1);
    H2=C3(ist:ist+131072-1);
    FC=20;
    nfftC=32768;
    windowC=hanning(nfftC);
    noverlapC=0.75*nfftC;
    dflagC='linear';

    for ichan=1:3
      if ichan==1

      [PZH1t,fC]=cohere(Z,H1,nfftC,FC,windowC,noverlapC,dflagC);
      PZH1=(PZH1t).^0.5;
 %     length(PZH1)=length(PZH1t);
 %     PZH1=PZH1(2:length(PZH1));
 %     fC=fC(2:length(PZH1));
```

```
%     NPD=64;
%     [ffZH1,PPZH1]=smooth_spec(fC,PZH1,NPD);

     I=Z;
     cloop=1

     elseif ichan==2

     [PH1H2t,fC]=cohere(H1,H2,nfftC,FC,windowC,noverlapC,dflagC);
     PH1H2=(PH1H2t).^0.5;
%     PH1H2=PH1H2(2:length(PH1H2));
%     fC=fC(2:length(PH1H2));
%     NPD=64;
%     [ffH1H2,PPH1H2]=smooth_spec(fC,PH1H2,NPD);

     I=H1;
     cloop=2

     elseif ichan==3

     [PZH2t,fC]=cohere(Z,H2,nfftC,FC,windowC,noverlapC,dflagC);
     PZH2=(PZH2t).^0.5;
%     PZH2=PZH2(2:length(PZH2));
%     fC=fC(2:length(PZH2));
%     NPD=64;
%     [ffZH2,PPZH2]=smooth_spec(fC,PZH2,NPD);

     I=H2;
     cloop=3
     end

   [PxxCt,fC]=psd(I,nfftC,FC,windowC,noverlapC,dflagC);
%    NN=length(fC);
%    PxxC=PxxCt(2:NN-1);
%    fC=fC(2:NN-1);
%    sintr=0.05;
%    NN=length(PxxC);
%    PxxC=PxxC.*sintr;
%    [mag,phase,fout,kend]=TF54000(nfftC/2);
%    PxxC = PxxC .* (mag.*10^(-9)).^2;
%    PxxC=PxxC(1:kend);
%    fC=fC(1:kend);
%    NPD=64;
%    NN=length(PxxC);
%    [ff,PP]=smooth_spec(fC,PxxC,NPD);
%    A=[ff,10*log10(PP)];
%    f=A(:,1);
%    SAdb=A(:,2);


     if ichan==1
     PxxCZ=PxxCt;
%     fZ=f;
     ploop=1
     elseif ichan==2
     PxxCH1=PxxCt;
%     fH1=f;
     ploop=2
     elseif ichan==3
     PxxCH2=PxxCt;
%     fH2=f;
```

```
      ploop=3
      end

    end

  transZH1=(PZH1).*((PxxCZ)./(PxxCH1)).^0.5;
  transH2H1=(PH1H2).*((PxxCH2)./(PxxCH1)).^0.5;
  transZH2=(PZH2).*((PxxCZ)./(PxxCH2)).^0.5;

  subplot(3,1,1), semilogx(fC,20.*log10(abs(transZH1)),'-');
  axis([10^-4 10^0 -75 25]);
  ylabel('magnitude (dB) ');
  TITLE=sprintf('Transfer Function plots H2O-Z and H2O-H1       CHUNK:
%d\n',ichunk);
  title(TITLE);
  fip = fopen(fileZH1(ichunk,:),'w+');
  fprintf (fip,' %f ',transZH1);
  fprintf (fip,'\n');
  fclose (fip);

  subplot(3,1,2), semilogx(fC,20.*log10(abs(transH2H1)),'-');
  axis([10^-4 10^0 -75 25]);
  ylabel('magnitude (dB) ');
  title('Transfer Function plots H2O-H1 and H2O-H2');
  fip = fopen(fileH2H1(ichunk,:),'w+');
  fprintf (fip,' %f ',transH2H1);
  fprintf (fip,'\n');
  fclose (fip);

  subplot(3,1,3), semilogx(fC,20.*log10(abs(transZH2)),'-');
  axis([10^-4 10^0 -75 25]);
  ylabel('magnitude (dB)');
  xlabel('frequency (Hz)');
  title('Transfer Function plots H2O-Z and H2O-H2');
  fip = fopen(fileZH2(ichunk,:),'w+');
  fprintf (fip,' %f ',transZH2);
  fprintf (fip,'\n');
  fclose (fip);

  orient landscape

  print
  end
```

- **t15_easytransf.m**

```
clf
clear

for ichunk=1:13

 FILE=sprintf('/dat1/u/zeldenru/OSN/spectra/H2Otest%2.2dH1H2.spec',ichunk);
  fileH1H2=(FILE);

  fid=fopen(fileH1H2,'rt');
  transH1H2=fscanf(fid,'%f');
  fclose(fid);

  FILE=sprintf('/dat1/u/zeldenru/OSN/spectra/H2Otest%2.2dZH1.spec',ichunk);
  fileZH1=(FILE);
```

```
  fid=fopen(fileZH1,'rt');
  transZH1=fscanf(fid,'%f');
  fclose(fid);

  FILE=sprintf('/dat1/u/zeldenru/OSN/spectra/H2Otest%2.2dZH2.spec',ichunk);
  fileZH2=(FILE);

  fid=fopen(fileZH2,'rt');
  transZH1=fscanf(fid,'%f');
  fclose(fid);


FILE=sprintf('/dat1/u/zeldenru/OSN/spectra/H2Otest%2.2dZH2c.spec',ichunk);
  fileZH2c=(FILE);

  fid=fopen(fileZH2c,'rt');
  transZH2c=fscanf(fid,'%f');
  fclose(fid);

  subplot(3,1,1), semilogx(fC,angle(transZH1),'-');
  ylabel('angle (rad) ');
  TITLE=sprintf('Transfer Function plots H2O-Z and H2O-H1       CHUNK:
%d\n',ichunk);
  title(TITLE);

  subplot(3,1,2), semilogx(fC,angle(transH2H1),'-');
  ylabel('angle (rad) ');
  title('Transfer Function plots H2O-H1 and H2O-H2');

  subplot(3,1,3), semilogx(fC,angle(transZH2),'-');
  ylabel('angle (rad)');
  xlabel('frequency (Hz)');
  title('Transfer Function plots H2O-Z and H2O-H2');
  orient landscape

  print
  end
```

- **t16_H2Otransfa.m**

```
clf
clear

%  load /osn1/b3s2/OSN/H2O/work/h20_121_HH

  kmm=length(C1);
  kmm=kmm/(131072);
  kmm=fix(kmm)

  for ichunk=1:kmm
    ronde=ichunk
    ist=(ichunk-1)*131072+1;
    Z=C1(ist:ist+131072-1);
    H1=C2(ist:ist+131072-1);
    H2=C3(ist:ist+131072-1);
    FC=20;
    nfftC=32768;
    windowC=hanning(nfftC);
    noverlapC=0.75*nfftC;
    dflagC='linear';
```

```
    for ichan=1:3
      if ichan==1

      [PZH1t,fC]=csd(Z,H1,nfftC,FC,windowC,noverlapC,dflagC);

%      length(PZH1)=length(PZH1t);
%      PZH1=PZH1(2:length(PZH1));
%      fC=fC(2:length(PZH1));
%      NPD=64;
%      [ffZH1,PPZH1]=smooth_spec(fC,PZH1,NPD);

      I=Z;
      cloop=1

      elseif ichan==2

      [PH1H2t,fC]=csd(H1,H2,nfftC,FC,windowC,noverlapC,dflagC);

%      PH1H2=PH1H2(2:length(PH1H2));
%      fC=fC(2:length(PH1H2));
%      NPD=64;
%      [ffH1H2,PPH1H2]=smooth_spec(fC,PH1H2,NPD);

      I=H1;
      cloop=2

      elseif ichan==3

      [PZH2t,fC]=csd(Z,H2,nfftC,FC,windowC,noverlapC,dflagC);

%      PZH2=PZH2(2:length(PZH2));
%      fC=fC(2:length(PZH2));
%      NPD=64;
%      [ffZH2,PPZH2]=smooth_spec(fC,PZH2,NPD);

      I=H2;
      cloop=3
      end

    [PxxCt,fC]=psd(I,nfftC,FC,windowC,noverlapC,dflagC);
%    NN=length(fC);
%    PxxC=PxxCt(2:NN-1);
%    fC=fC(2:NN-1);
%    sintr=0.05;
%    NN=length(PxxC);
%    PxxC=PxxC.*sintr;
%    [mag,phase,fout,kend]=TF54000(nfftC/2);
%    PxxC = PxxC .* (mag.*10^(-9)).^2;
%    PxxC=PxxC(1:kend);
%    fC=fC(1:kend);
%    NPD=64;
%    NN=length(PxxC);
%    [ff,PP]=smooth_spec(fC,PxxC,NPD);
%    A=[ff,10*log10(PP)];
%    f=A(:,1);
%    SAdb=A(:,2);


      if ichan==1
      PxxCZ=PxxCt;
```

```
 %     fZ=f;
      ploop=1
      elseif ichan==2
      PxxCH1=PxxCt;
 %     fH1=f;
      ploop=2
      elseif ichan==3
      PxxCH2=PxxCt;
 %     fH2=f;
      ploop=3
      end

    end

  PZH1=PZH1t./((PxxCZ.*PxxCH1).^0.5);
  PH1H2=PH1H2t./((PxxCH1.*PxxCH2).^0.5);
  PZH2=PZH2t./((PxxCZ.*PxxCH2).^0.5);

  transZH1=(PZH1).*((PxxCZ)./(PxxCH1)).^0.5;
  transH2H1=(PH1H2).*((PxxCH2)./(PxxCH1)).^0.5;
  transZH2=(PZH2).*((PxxCZ)./(PxxCH2)).^0.5;

  subplot(3,1,1), semilogx(fC,angle(transZH1),'-');
  axis([10^-4 10^0 -4 4]);
  ylabel('angle (rad) ');
  TITLE=sprintf('Transfer Function plots H2O-Z and H2O-H1       CHUNK:
%d\n',ichunk);
  title(TITLE);

  subplot(3,1,2), semilogx(fC,angle(transH2H1),'-');
  axis([10^-4 10^0 -4 4]);
  ylabel('angle (rad) ');
  title('Transfer Function plots H2O-H1 and H2O-H2');

  subplot(3,1,3), semilogx(fC,angle(transZH2),'-');
  axis([10^-4 10^0 -4 4]);
  ylabel('angle (rad)');
  xlabel('frequency (Hz)');
  title('Transfer Function plots H2O-Z and H2O-H2');
  orient landscape

  print
  end
```

- **t17s_OSNtransfa.m**

```
clf
clear

  fileH2H1(1,:)  = '../spectra/test01H1H2.spec';
      ect.


  nfftC=32768;

  fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BHZ');
  Zi=fread(fid,'float');
  fclose(fid);

  fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH1');
```

```
  H1i=fread(fid,'float');
  fclose(fid);

  fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH2');
  H2i=fread(fid,'float');
  fclose(fid);

  kmm=length(Zi);
  kmm=kmm/(131072);
  kmm=fix(kmm)

  for ichunk=1:kmm
    ronde=ichunk
    ist=(ichunk-1)*131072+1;
    Z=Zi(ist:ist+131072-1);
    H1=H1i(ist:ist+131072-1);
    H2=H2i(ist:ist+131072-1);
    FC=20;
    nfftC=32768;
    windowC=hanning(nfftC);
    noverlapC=0.75*nfftC;
    dflagC='linear';

    for ichan=1:3
      if ichan==1

      [PZH1t,fC]=csd(Z,H1,nfftC,FC,windowC,noverlapC,dflagC);

%      length(PZH1)=length(PZH1t);
%      PZH1=PZH1(2:length(PZH1));
%      fC=fC(2:length(PZH1));
%      NPD=64;
%      [ffZH1,PPZH1]=smooth_spec(fC,PZH1,NPD);

      I=Z;
      cloop=1

      elseif ichan==2

      [PH1H2t,fC]=csd(H1,H2,nfftC,FC,windowC,noverlapC,dflagC);

%      PH1H2=PH1H2(2:length(PH1H2));
%      fC=fC(2:length(PH1H2));
%      NPD=64;
%      [ffH1H2,PPH1H2]=smooth_spec(fC,PH1H2,NPD);

      I=H1;
      cloop=2

      elseif ichan==3

      [PZH2t,fC]=csd(Z,H2,nfftC,FC,windowC,noverlapC,dflagC);

%      PZH2=PZH2(2:length(PZH2));
%      fC=fC(2:length(PZH2));
%      NPD=64;
%      [ffZH2,PPZH2]=smooth_spec(fC,PZH2,NPD);

      I=H2;
      cloop=3
      end
```

```
    [PxxCt,fC]=psd(I,nfftC,FC,windowC,noverlapC,dflagC);
 %   NN=length(fC);
 %   PxxC=PxxCt(2:NN-1);
 %   fC=fC(2:NN-1);
 %   sintr=0.05;
 %   NN=length(PxxC);
 %   PxxC=PxxC.*sintr;
 %   [mag,phase,fout,kend]=TF54000(nfftC/2);
 %   PxxC = PxxC .* (mag.*10^(-9)).^2;
 %   PxxC=PxxC(1:kend);
 %   fC=fC(1:kend);
 %   NPD=64;
 %   NN=length(PxxC);
 %   [ff,PP]=smooth_spec(fC,PxxC,NPD);
 %   A=[ff,10*log10(PP)];
 %   f=A(:,1);
 %   SAdb=A(:,2);


      if ichan==1
      PxxCZ=PxxCt;
 %     fZ=f;
      ploop=1
      elseif ichan==2
      PxxCH1=PxxCt;
 %     fH1=f;
      ploop=2
      elseif ichan==3
      PxxCH2=PxxCt;
 %     fH2=f;
      ploop=3
      end

    end
%-------------------------------------------------------------------------
% This is the square root of the coherence, an imaginary array.
%-------------------------------------------------------------------------
  PZH1=PZH1t./((PxxCZ.*PxxCH1).^0.5);
  PH1H2=PH1H2t./((PxxCH1.*PxxCH2).^0.5);
  PZH2=PZH2t./((PxxCZ.*PxxCH2).^0.5);
%-------------------------------------------------------------------------
% This is the transferfunction, depending on the square root of the % % %
% coherence (imaginary) and the psd's of the two arrays.
%-------------------------------------------------------------------------
  transZH1=(PZH1).*((PxxCZ)./(PxxCH1)).^0.5;
  transH2H1=(PH1H2).*((PxxCH2)./(PxxCH1)).^0.5;
  transZH2=(PZH2).*((PxxCZ)./(PxxCH2)).^0.5;
%------------------------------------------------------------------------
% These are the FFT of the time series of the three channels, necessary for
% calculating noise in ChanH2 corrected for coherent noise in ChanH1 and %
% noise in ChanZ corrected for coherent noise in ChanH1.
%------------------------------------------------------------------------
  z=fft(Z);
  h1=fft(H1);
  h2=fft(H2);

  RZc=z-(transZH1.*h1);
  RH2c=h2-(transH1H2.*h1);
%-------------------------------------------------------------------------
% Converting back to time series by using inverse FFT
```

```
%-------------------------------------------------------------------------
  Zc=ifft(RZc)
  H2c=ifft(RH2c)
%-------------------------------------------------------------------------
% Calculating the csd and psd for Zc and H2c to calculate the corrected % %
% square root of coherence (imaginary) and with this the transfer function.
%-------------------------------------------------------------------------
  [PZH2tc,fC]=csd(Zc,H2c,nfftC,FC,windowC,noverlapC,dflagC);
  [PxxCH2c,fC]=psd(H2c,nfftC,FC,windowC,noverlapC,dflagC);
  [PxxCZc,fC]=psd(Zc,nfftC,FC,windowC,noverlapC,dflagC);
  PZH2c=PZH2tc./((PxxCZc.*PxxCH2c).^0.5);
  transZH2c=(PZH2c).*((PxxCZc./(PxxCH2c)).^0.5;
%-------------------------------------------------------------------------
% Calculating noise in ChanZc corrected for coherent noise in ChanH2c
%------------------------------------------------------------------------
  RZcc=RZc-(transZH2c.*RH2c);
  Zcc=ifft(RZcc)

pause

  subplot(3,1,1), semilogx(fC,(),'-');
  axis([10^-4 10^0 -4 4]);
  ylabel('angle (rad) ');
  TITLE=sprintf('Transfer Function plots OSN-Z and OSN-H1       CHUNK:
%d\n',ichunk);
  title(TITLE);
  fip = fopen(fileZH1(ichunk,:),'w+');
  fprintf (fip,' %f ',transZH1);
  fprintf (fip,'\n');
  fclose (fip);

  subplot(3,1,2), semilogx(fC,angle(transH2H1),'-');
  axis([10^-4 10^0 -4 4]);
  ylabel('angle (rad) ');
  title('Transfer Function plots OSN-H1 and OSN-H2');
  fip = fopen(fileH1H2(ichunk,:),'w+');
  fprintf (fip,' %f ',transH1H2);
  fprintf (fip,'\n');
  fclose (fip);

  subplot(3,1,3), semilogx(fC,angle(transZH2),'-');
  axis([10^-4 10^0 -4 4]);
  ylabel('angle (rad)');
  xlabel('frequency (Hz)');
  title('Transfer Function plots OSN-Z and OSN-H2');
  fip = fopen(fileZH2(ichunk,:),'w+');
  fprintf (fip,' %f ',transZH2);
  fprintf (fip,'\n');
  fclose (fip);

  orient landscape
  print
  end
```

- **t18_trytocorrectOSN.m**

```
clf
clear
```

```
fileH2H1(1,:)  = '../spectra/test01H1H2.spec';
    ect.


nfftC=32768;

fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BHZ');
Zi=fread(fid,'float');
fclose(fid);

fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH1');
H1i=fread(fid,'float');
fclose(fid);

fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH2');
H2i=fread(fid,'float');
fclose(fid);

kmm=length(Zi);
kmm=kmm/(131072);
kmm=fix(kmm)

for ichunk=1:kmm
  ronde=ichunk
  ist=(ichunk-1)*131072+1;
  Z=Zi(ist:ist+131072-1);
  H1=H1i(ist:ist+131072-1);
  H2=H2i(ist:ist+131072-1);
  FC=20;
  nfftC=32768;                %2^15
  windowC=hanning(nfftC);
  noverlapC=0.75*nfftC;
  dflagC='linear';
  delt=0.05;

  for ichan=1:3
    if ichan==1

    [PZH1t,fC]=csd(Z,H1,nfftC,FC,windowC,noverlapC,dflagC);

%     length(PZH1)=length(PZH1t);
%     PZH1=PZH1(2:length(PZH1));
%     fC=fC(2:length(PZH1));
%     NPD=64;
%     [ffZH1,PPZH1]=smooth_spec(fC,PZH1,NPD);

    I=Z;
    cloop=1

    elseif ichan==2

    [PH1H2t,fC]=csd(H1,H2,nfftC,FC,windowC,noverlapC,dflagC);

%     PH1H2=PH1H2(2:length(PH1H2));
%     fC=fC(2:length(PH1H2));
%     NPD=64;
%     [ffH1H2,PPH1H2]=smooth_spec(fC,PH1H2,NPD);

    I=H1;
    cloop=2
```

```
     elseif ichan==3

     [PZH2t,fC]=csd(Z,H2,nfftC,FC,windowC,noverlapC,dflagC);

%     PZH2=PZH2(2:length(PZH2));
%     fC=fC(2:length(PZH2));
%     NPD=64;
%     [ffZH2,PPZH2]=smooth_spec(fC,PZH2,NPD);

     I=H2;
     cloop=3
     end

   [PxxCt,fC]=psd(I,nfftC,FC,windowC,noverlapC,dflagC);
%   NN=length(fC);
%   PxxC=PxxCt(2:NN-1);
%   fC=fC(2:NN-1);
%   sintr=0.05;
%   NN=length(PxxC);
%   PxxC=PxxC.*sintr;
%   [mag,phase,fout,kend]=TF54000(nfftC/2);
%   PxxC = PxxC .* (mag.*10^(-9)).^2;
%   PxxC=PxxC(1:kend);
%   fC=fC(1:kend);
%   NPD=64;
%   NN=length(PxxC);
%   [ff,PP]=smooth_spec(fC,PxxC,NPD);
%   A=[ff,10*log10(PP)];
%   f=A(:,1);
%   SAdb=A(:,2);


     if ichan==1
     PxxCZ=PxxCt;
%     fZ=f;
     ploop=1
     elseif ichan==2
     PxxCH1=PxxCt;
%     fH1=f;
     ploop=2
     elseif ichan==3
     PxxCH2=PxxCt;
%     fH2=f;
     ploop=3
     end

   end

   PZH1=PZH1t./((PxxCZ.*PxxCH1).^0.5);
   PH1H2=PH1H2t./((PxxCH1.*PxxCH2).^0.5);
   PZH2=PZH2t./((PxxCZ.*PxxCH2).^0.5);
   transZH1=(PZH1).*((PxxCZ)./(PxxCH1)).^0.5;
   transH2H1=(PH1H2).*((PxxCH2)./(PxxCH1)).^0.5;
   transZH2=(PZH2).*((PxxCZ)./(PxxCH2)).^0.5;

   for itry=1:4
     het=(itry-1)*32768+1;
     zi=Z(het:het+32768-1);
     h1i=H1(het:het+32768-1);
     h2i=H2(het:het+32768-1);
     z(:,itry)=fft(zi);
```

```
      h1(:,itry)=fft(h1i);
      h2(:,itry)=fft(h2i);
    end


    pos_tf=transZH1;
    pos_f=(1/(delt*nfftC))*(0:nfftC/2);
      for index=(nfftC/2+2):nfftC;
        neg_f(index-(nfftC/2+1))=-pos_f(nfftC+2-index);
        neg_tf(index-(nfftC/2+1))=conj(-pos_tf(nfftC+2-index));
      end
    transZH1=[pos_tf neg_tf'];
    pos_tf=transH2H1;
    pos_f=(1/(delt*nfftC))*(0:nfftC/2);
      for index=(nfftC/2+2):nfftC;
        neg_f(index-(nfftC/2+1))=-pos_f(nfftC+2-index);
        neg_tf(index-(nfftC/2+1))=conj(-pos_tf(nfftC+2-index));
      end

    transH1H2=[pos_tf neg_tf'];
    total_f=[pos_f neg_f];
%     plot(total_f,transZH1);
%     plot(total_f,abs(transZH1));
    size(transZH1)
    size(transH2H1)


%     z=mean(mean (zi));                    %average of z1,z2,z3,z4.
%    h1=mean(mean (h1i));                   %average of h11,h12,h13,h14.
%    h2=mean(mean (h2i));                   %average of h1,h22,h23,h24.

    RZc1=z(:,1)-(transZH1.*h1(:,1));
    RZc2=z(:,2)-(transZH1.*h1(:,2));
    RZc3=z(:,3)-(transZH1.*h1(:,3));
    RZc4=z(:,4)-(transZH1.*h1(:,4));
    RH2c1=h2(:,1)-(transH2H1.*h1(:,1));
    RH2c2=h2(:,2)-(transH2H1.*h1(:,2));
    RH2c3=h2(:,3)-(transH2H1.*h1(:,3));
    RH2c4=h2(:,4)-(transH2H1.*h1(:,4));


    Zc1=ifft(RZc1);
    Zc2=ifft(RZc2);
    Zc3=ifft(RZc3);
    Zc4=ifft(RZc4);
    H2c1=ifft(RH2c1);
    H2c2=ifft(RH2c2);
    H2c3=ifft(RH2c3);
    H2c4=ifft(RH2c4);

    Zc=[Zc1 Zc2 Zc3 Zc4];
    H2c=[H2c1 H2c2 H2c3 H2c4];
    [PZH2tc,fC]=csd(Zc,H2c,nfftC,FC,windowC,noverlapC,dflagC);
    [PxxCH2c,fC]=psd(H2c,nfftC,FC,windowC,noverlapC,dflagC);
    [PxxCZc,fC]=psd(Zc,nfftC,FC,windowC,noverlapC,dflagC);
    PZH2c=PZH2tc./((PxxCZc.*PxxCH2c).^0.5);
    transZH2c=(PZH2c).*((PxxCZc)./(PxxCH2c)).^0.5;

    for itry=1:4
      het=(itry-1)*32768+1;
      Zi=Zc(het:het+32768-1);
```

```
      H2i=H2c(het:het+32768-1);
      z(:,itry)=fft(Zi);
      h2(:,itry)=fft(H2i);
    end

    pos_tf=transZH2c;
    index=(nfftC/2+2):nfftC;
    neg_tf(index-(nfftC/2+1))=conj(-pos_tf(nfftC+2-index));
    transZH2c=[pos_tf' neg_tf];
    size(transZH2c)

    RZcc1=RZc(:,1)-(transZH2c.*RH2c(:,1));
    RZcc2=RZc(:,2)-(transZH2c.*RH2c(:,2));
    RZcc3=RZc(:,3)-(transZH2c.*RH2c(:,3));
    RZcc4=RZc(:,4)-(transZH2c.*RH2c(:,4));

    Zcc1=ifft(RZcc1);
    Zcc2=ifft(RZcc2);
    Zcc3=ifft(RZcc3);
    Zcc4=ifft(RZcc4);
    Zcc=[Zcc1 Zcc2 Zcc3 Zcc4];
pause

    subplot(3,1,1), plot(H2c,'-');
%    axis([10^-4 10^0 -4 4]);
%    ylabel('angle (rad) ');
    TITLE=sprintf('Signal of OSN-H2 corrected for coh. with OSN-H1
CHUNK: %d\n',ichunk);
    title(TITLE);
%    fip = fopen(fileZH1(ichunk,:),'w+');
%    fprintf (fip,' %f ',transZH1);
%    fprintf (fip,'\n');
%    fclose (fip);

    subplot(3,1,2), plot(Zc,'-');
%    axis([10^-4 10^0 -4 4]);
%    ylabel('angle (rad) ');
    title('Signal of OSN-Z corrected for coh. with OSN-H1');
%    fip = fopen(fileH1H2(ichunk,:),'w+');
%    fprintf (fip,' %f ',transH1H2);
%    fprintf (fip,'\n');
%    fclose (fip);

    subplot(3,1,3), plot(Zcc,'-');
%    axis([10^-4 10^0 -4 4]);
%    ylabel('angle (rad)');
    xlabel('time (?)');
    title('Signal of OSN-Z corrected for coh. with OSN-H1 and OSN-H2');
%    fip = fopen(fileZH2(ichunk,:),'w+');
%    fprintf (fip,' %f ',transZH2);
%    fprintf (fip,'\n');
%    fclose (fip);

    orient landscape
%    print
  end
```

- **t19_totalcorrectOSN.m**

```
clf
```

```
clear

%----OPEN FILES WITH DATA OF CHANNELS Z,H1,H2-------------------------------

  nfftC=32768;

  fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BHZ');
  Zi=fread(fid,'float');
  fclose(fid);

  fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH1');
  H1i=fread(fid,'float');
  fclose(fid);

  fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH2');
  H2i=fread(fid,'float');
  fclose(fid);

%----READ DATA IN CHUNKS OF 1.8 HOURS--------------------------------------

  kmm=length(Zi);
  kmm=kmm/(131072);
  kmm=fix(kmm)

  for ichunk=1:kmm
    ronde=ichunk
    ist=(ichunk-1)*131072+1;

    Z=Zi(ist:ist+131072-1);
    H1=H1i(ist:ist+131072-1);
    H2=H2i(ist:ist+131072-1);

%----PLOTTING TIME SERIES OF ORIGINAL Z-------------------------------------

    Zplot=Zi(ist:1500:ist+131072-1);
%    H1plot=H1i(ist:1500:ist+131072-1);
%    H2plot=H2i(ist:1500:ist+131072-1);

    subplot('position',[0.1500 0.7475 0.8000 0.1725 ])
    plot(Zplot,'-');
    grid on
    TITLE=sprintf('Signal of OSN-Z       CHUNK: %d\n',ichunk);
    title(TITLE);
    axis([0 90 -90000 -70000]);
%    subplot(5,1,1), plot(H1p,'-');
%    title('Signal of OSN-H1');

%    subplot(5,1,1), plot(H2p,'-');
%    title('Signal of OSN-H2');
%    axis([0 90 -40000 20000]);
%    pause

%----SET PARAMETERS---------------------------------------------------------

    FC=20;
    nfftC=32768;                %2^15
    windowC=hanning(nfftC);
    noverlapC=0.75*nfftC;
    dflagC='linear';
    delt=0.05;
```

```
%------CALCULATING CSD------------------------------------------------------

    [ZH1csd,fC]=csd(Z,H1,nfftC,FC,windowC,noverlapC,dflagC);

 %       length(PZH1)=length(PZH1t);
 %       PZH1=PZH1(2:length(PZH1));
 %       fC=fC(2:length(PZH1));
 %       NPD=64;
 %       [ffZH1,PPZH1]=smooth_spec(fC,PZH1,NPD);


    [H2H1csd,fC]=csd(H2,H1,nfftC,FC,windowC,noverlapC,dflagC);

 %       PH1H2=PH1H2(2:length(PH1H2));
 %       fC=fC(2:length(PH1H2));
 %       NPD=64;
 %       [ffH1H2,PPH1H2]=smooth_spec(fC,PH1H2,NPD);
        cloop=2

    [ZH2csd,fC]=csd(Z,H2,nfftC,FC,windowC,noverlapC,dflagC);

 %       PZH2=PZH2(2:length(PZH2));
 %       fC=fC(2:length(PZH2));
 %       NPD=64;
 %       [ffZH2,PPZH2]=smooth_spec(fC,PZH2,NPD);

%----CALCULATING PSD (H2,H1,Z) AND SMOOTHED PSD (Z)-----------------------

    [PxxCH2,fC]=psd(H2,nfftC,FC,windowC,noverlapC,dflagC);
    [PxxCH1,fC]=psd(H1,nfftC,FC,windowC,noverlapC,dflagC);
    [PxxCZ,fC]=psd(Z,nfftC,FC,windowC,noverlapC,dflagC);
    NN=length(fC);
    PxxC=PxxCZ(2:NN-1);
    fC=fC(2:NN-1);
    sintr=0.05;
    NN=length(PxxC);
    PxxC=PxxC.*sintr;
    [mag,phase,fout,kend]=TF54000(nfftC/2);
    PxxC = PxxC .* (mag.*10^(-9)).^2;
    PxxC=PxxC(1:kend);
    fC=fC(1:kend);
    NPD=64;
    NN=length(PxxC);
    [ff,PP]=smooth_spec(fC,PxxC,NPD);
    A=[ff,10*log10(PP)];
    sfC=A(:,1);
    sPxxCZ=A(:,2);

%----CALCULATING THE COMPLEX (ROOT OF) THE COHERENCE FUNCTION--------------

    PZH1=ZH1csd./((PxxCZ.*PxxCH1).^0.5);
    PH1H2=H2H1csd./((PxxCH2.*PxxCH1).^0.5);
    PZH2=ZH2csd./((PxxCZ.*PxxCH2).^0.5);

%----CALCULATING THE TRANSFER FUNCTIONS------------------------------------

    transZH1=PZH1.*((PxxCZ./PxxCH1).^0.5);
    transH2H1=PH2H1.*((PxxCH2./PxxCH1).^0.5);
    transZH2=PZH2.*((PxxCZ./PxxCH2).^0.5);
```

```
%----DIVIDING THE ARRAYS (Z,H1,H2) IN 4 PARTS TO GET LENGTH(FFT) EQUAL TO
LENGTH(TRANSF)=32768------------------------

    for itry=1:4
      het=(itry-1)*32768+1;
      zi=Z(het:het+32768-1);
      h1i=H1(het:het+32768-1);
      h2i=H2(het:het+32768-1);
      z(:,itry)=fft(zi);
      h1(:,itry)=fft(h1i);
      h2(:,itry)=fft(h2i);
    end

%----GIVING THE ARRAY'S OF (TRANSF,F) A NEGATIVE PART, THEREBY DOUBLING
THEIR LENGTH=32768----------------------

    pos_tf=transZH1;
    pos_f=(1/(delt*nfftC))*(0:nfftC/2);
      for index=(nfftC/2+2):nfftC;
        neg_f(index-(nfftC/2+1))=-pos_f(nfftC+2-index);
        neg_tf(index-(nfftC/2+1))=conj(-pos_tf(nfftC+2-index));
      end
    nega_tf=(neg_tf');
    transZH1=[pos_tf;nega_tf];
    total_fZH1=[pos_f neg_f];
%     plot(total_fZH1,transZH1);
%     plot(total_fZH1,abs(transZH1));
    ind=find(abs(total_f)>0.1);
    transZH1(ind)=0.0;

    pos_tf=transH2H1;
    pos_f=(1/(delt*nfftC))*(0:nfftC/2);
      for index=(nfftC/2+2):nfftC
        neg_f(index-(nfftC/2+1))=-pos_f(nfftC+2-index);
        neg_tf(index-(nfftC/2+1))=conj(-pos_tf(nfftC+2-index));
      end
    nega_tf=(neg_tf');
    transH2H1=[pos_tf;nega_tf];
    total_fH2H1=[pos_f neg_f];
%     plot(total_fH2H1,transH2H1);
%     plot(total_fH2H1,abs(transH2H1));
    ind=find(abs(total_f)>0.1);
    transH2H1(ind)=0.0;

    size(transZH1)
    size(transH2H1)

%----CALCULATING THE CORRECTED VALUES (RZc & RH2c) IN 4 PARTS--------------

    RZc1=z(:,1)-(transZH1.*h1(:,1));
    RZc2=z(:,2)-(transZH1.*h1(:,2));
    RZc3=z(:,3)-(transZH1.*h1(:,3));
    RZc4=z(:,4)-(transZH1.*h1(:,4));
    RH2c1=h2(:,1)-(transH2H1.*h1(:,1));
    RH2c2=h2(:,2)-(transH2H1.*h1(:,2));
    RH2c3=h2(:,3)-(transH2H1.*h1(:,3));
    RH2c4=h2(:,4)-(transH2H1.*h1(:,4));

%----CONVERTING BACK FROM FFT TO TIME SERIES--------------------------------

    Zc1=ifft(RZc1);
```

```
    Zc2=ifft(RZc2);
    Zc3=ifft(RZc3);
    Zc4=ifft(RZc4);
    H2c1=ifft(RH2c1);
    H2c2=ifft(RH2c2);
    H2c3=ifft(RH2c3);
    H2c4=ifft(RH2c4);

%----ADDITION OF 4 PARTS TO FORM ONE CHUNK AGAIN---------------------------

    Zc=[Zc1;Zc2;Zc3;Zc4];
    H2c=[H2c1;H2c2;H2c3;H2c4];

%----CALCULATING CSD, PSD, COMPLEX (ROOT OF) THE COHERENCE FUNCTION,
TRANSFER FUNCTIONS---------------
%--------------------------FOR CORRECTED Z & H2 (Zc&H2c)--------------------

    [ZH2csdc,fC]=csd(abs(Zc),abs(H2c),nfftC,FC,windowC,noverlapC,dflagC);

    [PxxCH2c,fC]=psd(abs(H2c),nfftC,FC,windowC,noverlapC,dflagC);
    [PxxCZc,fC]=psd(abs(Zc),nfftC,FC,windowC,noverlapC,dflagC);

%----IN CASE WE WANT TO PLOT SMOOTHED CORRECTED PSD'S----------------------
    for nu=1:2
    NN=length(fC);
      if nu ==1
      PxxC=PxxCH2c;
      elseif nu ==2
      PxxC=PxxCZc;
      end
    PxxC=PxxCt(2:NN-1);
    fC=fC(2:NN-1);
    sintr=0.05;
    NN=length(PxxC);
    PxxC=PxxC.*sintr;
    [mag,phase,fout,kend]=TF54000(nfftC/2);
    PxxC = PxxC .* (mag.*10^(-9)).^2;
    PxxC=PxxC(1:kend);
    fC=fC(1:kend);
    NPD=64;
    NN=length(PxxC);
    [ff,PP]=smooth_spec(fC,PxxC,NPD);
    A=[ff,10*log10(PP)];
      if nu==1
      fCn=A(:,1);
      PxxCH2cn=A(:,2);
      elseif nu==2
      fCnu=A(:,1);
      PxxCZcn=A(:,2);
      end
    end

%-------------------------------------------------------------------------

    PZH2c=PZH2tc./((PxxCZc.*PxxCH2c).^0.5);
    transZH2c=(PZH2c).*((PxxCZc)./(PxxCH2c)).^0.5;

%----DIVIDING THE ARRAYS (Zc,H2c) IN 4 PARTS TO GET LENGTH(FFT) EQUAL TO
LENGTH(TRANSF)=32768---------

    for itried=1:4
```

```
      hets=(itried-1)*32768+1;
      zi=Zc(hets:hets+32768-1);
      h2i=H2c(hets:hets+32768-1);
      zc(:,itried)=fft(zi);
      h2c(:,itried)=fft(h2i);
    end

%----GIVING THE ARRAY'S OF (TRANSF,F) A NEGATIVE PART, THEREBY DOUBLING
THEIR LENGTH=32768------------

    pos_tf=transZH2c;
    index=(nfftC/2+2):nfftC;
    neg_tf(index-(nfftC/2+1))=conj(-pos_tf(nfftC+2-index));
    nega_tf=(neg_tf');
    transZH2c=[pos_tf;nega_tf];
    size(transZH2c)

%----CALCULATING THE CORRECTED VALUES (RZcc) IN 4 PARTS--------------------
    RZcc1=zc(:,1)-(transZH2c.*h2c(:,1));
    RZcc2=zc(:,2)-(transZH2c.*h2c(:,2));
    RZcc3=zc(:,3)-(transZH2c.*h2c(:,3));
    RZcc4=zc(:,4)-(transZH2c.*h2c(:,4));

%----CONVERTING BACK FROM FFT TO TIME SERIES & ADDITION OF 4 PARTS TO FORM
ONE CHUNK AGAIN-----------------------------------------------------------

    Zcc1=ifft(RZcc1);
    Zcc2=ifft(RZcc2);
    Zcc3=ifft(RZcc3);
    Zcc4=ifft(RZcc4);
    Zcc=[Zcc1;Zcc2;Zcc3;Zcc4];

%----SMOOTHED CORRECTED PSD (Zcc)-------------------------------------

    [PxxCZcc,fC]=psd(abs(Zcc),nfftC,FC,windowC,noverlapC,dflagC);
    NN=length(fC);
    PxxC=PxxCZcc(2:NN-1);
    fC=fC(2:NN-1);
    sintr=0.05;
    NN=length(PxxC);
    PxxC=PxxC.*sintr;
    [mag,phase,fout,kend]=TF54000(nfftC/2);
    PxxC = PxxC .* (mag.*10^(-9)).^2;
    PxxC=PxxC(1:kend);
    fC=fC(1:kend);
    NPD=64;
    NN=length(PxxC);
    [ff,PP]=smooth_spec(fC,PxxC,NPD);
    A=[ff,10*log10(PP)];
    fCnut=A(:,1);
    PxxCZccn=A(:,2);

%----MULTIPLY BY -1, BECAUSE PLOTTING SHOWS OVERTURNED CURVES--------------

    H2cp=-1*abs(H2c(1:1500:131072));
    Zcp=-1*abs(Zc(1:1500:131072));
    Zccp=-1*abs(Zcc(1:1500:131072));

%----PLOTTING--------------------------------------------------------------

%    subplot(5,1,2), plot(H2cp,'-');
```

```
%    axis([0 90 -40000 20000]);
%    TITLE=sprintf('Signal of OSN-H2 corrected for coh. with OSN-H1
CHUNK: %d\n',ichunk);
%    title(TITLE);

    grid on
    subplot('position',[0.1500 0.525 0.8000 0.1725  ])
    plot(Zcp,'-');
    axis([0 90 -90000 -70000]);
%    ylabel('angle (rad) ');
    title('Signal of OSN-Z corrected for coh. with OSN-H1');

    grid on
    subplot('position',[0.1500 0.3025 0.8000 0.1725 ])
    plot(Zccp,'-');
    axis([0 90 -90000 -70000]);
%    ylabel('angle (rad)');
    xlabel('time (?)');
    title('Signal of OSN-Z corrected for coh. with OSN-H1 and OSN-H2');

    grid off
    subplot('position',[0.1500 0.08 0.8000 0.1725 ])
    semilogx (fZ,SAdbZ,'m--')
    hold on
%    semilogx (fCnu,PxxCZcn,'b-')
%    hold on
%    semilogx(fCnut,PxxCZccn,'r-.')
%    hold on
    semilogx(fCn,PxxCH2cn,'c:')
    xlabel('frequency (Hz)');
    ylabel('psd');
    title('Corrected Power Spectral Density
');
    hold on

%----LAYOUT PLOTTING----------------------------------------------------

    shift=50;
    x=[1.2*10^(-3) 1.2*10^(-3) 0.8*10^(-1) 0.8*10^(-1)];
    y=[-163+shift -105+shift -105+shift -163+shift];
    patch(x,y,'w')
    text(2*10^(-3),-111+shift,'OSN-Z-H1           ');
    text(2*10^(-3),-126+shift,'OSN-Z-H1           ');
    text(2*10^(-3),-142+shift,'OSN-Z-H1-H2       ');
    text(2*10^(-3),-157+shift,'OSN-H2-H1    ');
    plot([0.3*10^(-1) 0.6*10^(-1)],[-111+shift -111+shift],'m--')
    plot([0.3*10^(-1) 0.6*10^(-1)],[-126+shift -126+shift],'b-')
    plot([0.3*10^(-1) 0.6*10^(-1)],[-142+shift -142+shift],'r-.')
    plot([0.3*10^(-1) 0.6*10^(-1)],[-157+shift -157+shift],'c:')

    for m=1:2
      if m==1
      eval(['load ../matlab/nlnm_avd ;'])
      eval(['NM=nlnm_avd;']);
      elseif m==2
      eval(['load ../matlab/nhnm_avd ;'])
      eval(['NM=nhnm_avd;']);
      end
    km=1:length(NM);
    tm  =NM(km,1);        % periods(sec)
    fm  =NM(km,2);        % frequency(Hz)
```

```
    acc =NM(km,3);      % db refered to acceleration
    vel =NM(km,4);      % db refered to velocity
    dis =NM(km,5);      % db refered to displacement
    semilogx(fm,acc,'k--')
    axis([10^-4 10^1 -200 -40]);
    end
    hold off
    set (gcf,'paperposition',[0 0 8.5 10.8],'paperorientation','portrait')
    pause
%    print

  end
```

- **t20_testnorm.m**

```
clf
clear

  FC=20;
  nfftC=32768;               %2^15
  windowC=hanning(nfftC);
  noverlapC=0.75*nfftC;
  dflagC='none';
  delt=0.05;
  NN=131072;

  t=delt.*(1:NN);
  pos_f=(1/(delt*nfftC))*(0:nfftC/2);
  for index=(nfftC/2+2):nfftC
    neg_f(index-(nfftC/2+1))=-pos_f(nfftC+2-index);
  end
  total_f=[pos_f neg_f];

  fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BHZ');
  Zi=fread(fid,131072,'float');
  fclose(fid);

  fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH2');
  H2i=fread(fid,131072,'float');
  fclose(fid);

  Z=Zi;
  H2=H2i;

  plot(t,Z,'b-',t,H2,'r-')
  title('Vertical (bleu) and Horizontal (red) Time Series')
  pause
  hold off

  [PxxCZ,fC]=psd(Z,nfftC,FC,windowC,noverlapC,dflagC);
  [PxxCH2,fC]=psd(H2,nfftC,FC,windowC,noverlapC,dflagC);

  semilogx(fC,10.*log10(PxxCZ),'b-',fC,10.*log10(PxxCH2),'r-')
  title('Vertical (blue) and Horizontal (red) PSD from Matlab')
  hold on
  pause

  nstep=nfftC-noverlapC;
  newPZ=zeros(1,nfftC)';
  for index=1:13
```

```
    istart=(index-1)*nstep+1;
    iend=istart+nfftC-1;
    fftbank(:,index)=fft(windowC.*Z(istart:iend));
    newPZ=newPZ+abs(fftbank(:,index)).^2;
  end
  newPZ=newPZ/(norm(windowC)^2*13);

  semilogx((0:(nfftC-1))/nfftC*FC,10*log10(newPZ),'k.');

  hold off
```

- **t21_OSNwindcorZ.m**

```
clf
clear

%----------------------------------------------------------------------

  fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BHZ');
  Zi=fread(fid,131072,'float');
  fclose(fid);

  fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH1');
  H1i=fread(fid,131072,'float');
  fclose(fid);

  fid=fopen('../osn1_compressed/1998/096/19980960000.00.OSN1.BH2');
  H2i=fread(fid,131072,'float');
  fclose(fid);

%--------------------------------------------------------------------------

  kmm=length(Zi);
  kmm=kmm/(131072);
  kmm=fix(kmm)

  for ichunk=1:kmm
    ronde=ichunk
    ist=(ichunk-1)*131072+1;

    Z=Zi(ist:ist+131072-1);
    H1=H1i(ist:ist+131072-1);
    H2=H2i(ist:ist+131072-1);

    FC=20;
    nfftC=32768;                %2^15
    windowC=hanning(nfftC);
    noverlapC=0.75*nfftC;
    dflagC='none';
    delt=0.05;


%------CALCULATING CSD-----------------------------------------------------

    [ZH1csd,fC]=csd(Z,H1,nfftC,FC,windowC,noverlapC,dflagC);

    [H2H1csd,fC]=csd(H2,H1,nfftC,FC,windowC,noverlapC,dflagC);

    [ZH2csd,fC]=csd(Z,H2,nfftC,FC,windowC,noverlapC,dflagC);
```

```
%----CALCULATING PSD (H2,H1,Z) AND SMOOTHED PSD (Z)------------------------

    [PxxCH2,fC]=psd(H2,nfftC,FC,windowC,noverlapC,dflagC);
    [PxxCH1,fC]=psd(H1,nfftC,FC,windowC,noverlapC,dflagC);
    [PxxCZ,fCi]=psd(Z,nfftC,FC,windowC,noverlapC,dflagC);
    NN=length(fCi)
    PxxC=PxxCZ(2:NN-1);
    fC=fCi(2:NN-1);
    sintr=0.05;
    NN=length(PxxC)
    PxxC=PxxC.*sintr;
    [mag,phase,fout,kend]=TF54000(nfftC/2);
    PxxC = PxxC .* (mag.*10^(-9)).^2;
    PxxC=PxxC(1:kend);
    fC=fC(1:kend);
    NPD=64;
    NN=length(PxxC)
    [ff,PP]=smooth_spec(fC,PxxC,NPD);
    A=[ff,10*log10(PP)];
    sfC=A(:,1);
    sPxxCZ=A(:,2);
%    pause

%----CALCULATING THE COMPLEX (ROOT OF) THE COHERENCE FUNCTION--------------

    PZH1=ZH1csd./((PxxCZ.*PxxCH1).^0.5);
    PH2H1=H2H1csd./((PxxCH2.*PxxCH1).^0.5);
    PZH2=ZH2csd./((PxxCZ.*PxxCH2).^0.5);

%----CALCULATING THE TRANSFER FUNCTIONS------------------------------------

    transZH1=PZH1.*((PxxCZ./PxxCH1).^0.5);
    transH2H1=PH2H1.*((PxxCH2./PxxCH1).^0.5);
    transZH2=PZH2.*((PxxCZ./PxxCH2).^0.5);

%----GIVING THE ARRAY'S OF (TRANSF,F) A NEGATIVE PART, THEREBY DOUBLING
THEIR LENGTH=32768----------------------

    pos_tf=transZH2;
    pos_f=(1/(delt*nfftC))*(0:nfftC/2);
      for index=(nfftC/2+2):nfftC;
        neg_f(index-(nfftC/2+1))=-pos_f(nfftC+2-index);
        neg_tf(index-(nfftC/2+1))=conj(-pos_tf(nfftC+2-index));
      end
    nega_tf=(neg_tf');
    transZH2=[pos_tf;nega_tf];
    total_fZH2=[pos_f neg_f];
%     plot(total_f,transZH1);
%     plot(total_f,abs(transZH1));
    ind=find(abs(total_fZH2)>0.1);
    transZH1(ind)=0.0;

%----CALCULATING THE FFT OVER 13 OVERLAPPING HANNING WINDOWS WITH OVERLAP
(Z&H2)---------

    nstep=nfftC-noverlapC;
    newPZ=zeros(1,nfftC)';
    for index=1:13
      istart=(index-1)*nstep+1;
      iend=istart+nfftC-1;
      fftbank(:,index)=fft(windowC.*Z(istart:iend));
```

```
      newPZ=newPZ+abs(fftbank(:,index)).^2;
    end
    newPZ=newPZ/(norm(windowC)^2*13);

    newPH1=zeros(1,nfftC)';
    for index=1:13
      istart=(index-1)*nstep+1;
      iend=istart+nfftC-1;
      fftbank(:,index)=fft(windowC.*H1(istart:iend));
      newPH1=newPH1+abs(fftbank(:,index)).^2;
    end
    newPH1=newPH1/(norm(windowC)^2*13);

%     semilogx((0:(nfftC-1))/nfftC*FC,10*log10(newPZ),'k.');


%----CALCULATING THE CORRECTED VALUES (RZc)--------------------------------

    RZc1=newPZ-(transZH2.*newPH1);

%----SMOOTHED CORRECTED PSD (RZc)----------------------------------------

%     [PxxCZcc,fC]=psd(abs(Zcc),nfftC,FC,windowC,noverlapC,dflagC);

    RZc1m=abs(RZc1(1:nfftC/2+1));
    NN=length(fCi)
    PxxC=RZc1m(2:NN-1);
    fC=fCi(2:NN-1);
    sintr=0.05;
    NN=length(PxxC)
    PxxC=PxxC.*sintr;
    [mag,phase,fout,kend]=TF54000(nfftC/2);
    PxxC = PxxC .* (mag.*10^(-9)).^2;
    PxxC=PxxC(1:kend);
    fC=fC(1:kend);
    NPD=64;
    NN=length(PxxC)
    [ff,PP]=smooth_spec(fC,PxxC,NPD);
    A=[ff,10*log10(PP)];
    f=A(:,1);
    sRZc=A(:,2);

%----PLOTTING-------------------------------------------------------------

    semilogx (sfC,sPxxCZ,'b-.')
    hold on
    semilogx(f,sRZc,'r-')
    xlabel('frequency (Hz)');
    ylabel('psd');
    title('Corrected Power Spectral Density
');
    hold on

%----LAYOUT PLOTTING------------------------------------------------------

    shift=50;
    x=[1.2*10^(-3) 1.2*10^(-3) 0.8*10^(-1) 0.8*10^(-1)];
    y=[-163+shift -105+shift -105+shift -163+shift];
    patch(x,y,'w')
%     text(2*10^(-3),-111+shift,'OSN-Z-H1           ');
    text(2*10^(-3),-126+shift,'OSN-Z           ');
```

```
    text(2*10^(-3),-142+shift,'OSN-Z-H2         ');
%    text(2*10^(-3),-157+shift,'OSN-H2-H1  ');
%    plot([0.3*10^(-1) 0.6*10^(-1)],[-111+shift -111+shift],'m--')
    plot([0.3*10^(-1) 0.6*10^(-1)],[-126+shift -126+shift],'b-.')
    plot([0.3*10^(-1) 0.6*10^(-1)],[-142+shift -142+shift],'r-')
%    plot([0.3*10^(-1) 0.6*10^(-1)],[-157+shift -157+shift],'c:')

    for m=1:2
      if m==1
      eval(['load ../matlab/nlnm_avd ;'])
      eval(['NM=nlnm_avd;']);
      elseif m==2
      eval(['load ../matlab/nhnm_avd ;'])
      eval(['NM=nhnm_avd;']);
      end
    km=1:length(NM);
    tm  =NM(km,1);        % periods(sec)
    fm  =NM(km,2);        % frequency(Hz)
    acc =NM(km,3);      % db refered to acceleration
    vel =NM(km,4);      % db refered to velocity
    dis =NM(km,5);      % db refered to displacement
    semilogx(fm,acc,'k--')
    axis([10^-4 10^1 -200 -40]);
    end
    hold off
    set (gcf,'paperposition',[0 0 8.5 10.8],'paperorientation','portrait')
%   pause
%    print

  end
```

- **smooth_spec.m**

```
% function [fnew,Pnew]= smooth_spec(f,P,NPD);
% smooth spectra to approximatly even spacing in log f
% USAGE: [fnew,Pnew]= smooth_spec(f,P,NPD);
% INPUT:
%    f column vector of frequencies (Hz)
%    P column matrix of one or more power spectra
%    NPD scalar specifying the number of frequencies per decade on output
%
% OUTPUT
%    fnew column vector containing the new frequencies
%    Pnew column matrix containing the smoothed power spectra
%
% Divide the frequency band into distinct regions that are evenly spaced in
% log10 f, with NPD frequencies per decade.  Within each of these regions %
% take the median of all power estimates to get Pnew and the median of all
% the frequencies to get fnew.

Fmin=floor(NPD*log10(min(f))-1)/NPD;
Fmax= ceil(NPD*log10(max(f))+1)/NPD;
Fnew=10.^(Fmin:1/NPD:Fmax)';           % New target frequencies
Fbound=10.^((Fmin-1/NPD:1/NPD:Fmax)+.5/NPD)';%Boundaries between new
frequencies
Nnew=length(Fnew);                     % Number of new periods
fnew=Fnew*0;Pnew=P*0;k=0;              % initialize vectors
for i=1:Nnew
  ind=find(f>Fbound(i) & f<=Fbound(i+1)); % index of spectra in the ith
interval
```

```
  N=length(ind);                        % number of points in ith interval
  if N==1,                              % if only one point leave it and its
old frequency
    k=k+1;
    fnew(k)=f(ind);
    Pnew(k,:)=P(ind,:);
  elseif N>1                            % if more than one frequency take
median of all
    k=k+1;                              % points in the interval and set the
frequency to
    fnew(k)=median(f(ind));             % be the median of the frequencies
    Pnew(k,:)=median(P(ind,:));
  end
end
fnew=fnew(1:k);
Pnew=Pnew(1:k,:);
```

Appendix F

- **r01_FC160psdH2O***.m**

```
clf
clear

filename=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day091_160.frequency'
)
fid=fopen(filename,'rt');
f=fscanf(fid,' %f \n');
fclose(fid);

%  for ichunk=1:13
    ichunk=1

FILE=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day091_160_%2.2d_Z.spectr
a',ichunk);
    fileZ=(FILE);
    fid=fopen(fileZ,'rt');
    Z=fscanf(fid,' %f \n');
    fclose(fid);
    semilogx(f,Z,'c-');
    TITLE=sprintf('PSD plots Z(-), H1(--), H2(-.) of H2O data at FC=160
CHUNK: %d\n',ichunk);
    title(TITLE);
    axis([10^-1 10^2 -165 -75]);
%    hold on


FILE=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day091_160_%2.2d_H1.spect
ra',ichunk);
    fileH1=(FILE);
    fid=fopen(fileH1,'rt');
    H1=fscanf(fid,' %f \n');
    fclose(fid);
    semilogx(f,H1,'r-');
    axis([10^-1 10^2 -165 -75]);
%    hold on


FILE=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day091_160_%2.2d_H2.spect
ra',ichunk);
    fileH2=(FILE);
    fid=fopen(fileH2,'rt');
    H2=fscanf(fid,' %f \n');
    fclose(fid);
    semilogx(f,H2,'b-');
    axis([10^-1 10^2 -165 -75]);
%    hold on
%    pause
%   print
    hold off
% end
```

- **r02_FC160tomH2O***.m**

```
clf
clear
```

```
filename=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day121_160.frquency')
fid=fopen(filename,'rt');
f=fscanf(fid,' %f \n');
fclose(fid);

%  for ichunk=1:13
    ichunk=1

FILE=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day121_%2.2d_C1.spectra',
ichunk);
    fileZ=(FILE);
    fid=fopen(fileZ,'rt');
    Z=fscanf(fid,' %f \n');
    fclose(fid);
    semilogx(f,Z,'c-');
    grid on
    TITLE=sprintf('PSD plots Z(-), H1(--), H2(-.) of H2O data at FC=160
CHUNK: %d\n',ichunk);
    title(TITLE);
    axis([0.9 60 -165 -75]);
%    hold on


FILE=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day121_%2.2d_C2.spectra',
ichunk);
    fileH1=(FILE);
    fid=fopen(fileH1,'rt');
    H1=fscanf(fid,' %f \n');
    fclose(fid);
    semilogx(f,H1,'r-');
    grid on
    axis([0.9 60 -165 -75]);
%    hold on


FILE=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day121_%2.2d_C3.spectra',
ichunk);
    fileH2=(FILE);
    fid=fopen(fileH2,'rt');
    H2=fscanf(fid,' %f \n');
    fclose(fid);
    semilogx(f,H2,'b-');
 %   plot (f,H2,'b-');
    grid on
    axis([0.9 60 -165 -75]);
%    hold on
%    pause
%   print
    hold off
% end
```

- **r03_polyfit5H2O***.m**

```
clf
clear

filename=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day091_160.frequency'
);
  fid=fopen(filename,'rt');
```

```
  f=fscanf(fid,' %f \n');
  fclose(fid);

  for ichunk=1:13
%    ichunk=1
    for ichan=2:3
      if ichan==1

FILE=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day091_160_%2.2d_Z.spectr
a',ichunk)
        fileZ=(FILE);

        fid=fopen(fileZ,'rt');
        Z=fscanf(fid,' %f \n');
        fclose(fid);
        semilogx(f,Z,'r-');
        axis([0.9 70 -165 -75]);
        TITLE=sprintf('PSD plots Z(-), H1(--), H2(:) of H2O data at FC=160
CHUNK: %d\n',ichunk);
        title(TITLE);
        [freqZ,powerZ]=ginput(5)
%        hold on
      elseif ichan==2

FILE=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day091_160_%2.2d_H1.spect
ra',ichunk)
        fileH1=(FILE);

        fid=fopen(fileH1,'rt');
        H1=fscanf(fid,' %f \n');
        fclose(fid);
        semilogx(f,H1,'r-');
        axis([0.9 70 -185 -95]);
%        [freqH1,powerH1]=ginput(5)
        hold on
      elseif ichan==3

FILE=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day091_160_%2.2d_H2.spect
ra',ichunk)
        fileH2=(FILE);

        fid=fopen(fileH2,'rt');
        H2=fscanf(fid,' %f \n');
        fclose(fid);
        semilogx(f,H2,'b-');
        axis([0.9 70 -185 -95]);
        [freqH2,powerH2]=ginput(5)
        hold off
      end
    end
%    pause


    n=[1 2 3 4 5];

%    freqH1=freqH1';
    freqH2=freqH2';

%    [FH1i,bH1i]=polyfit(n,freqH1,1);
%    FH1=FH1i(1)
%    bH1=FH1i(2)
```

```
    [FH2i,bH2i]=polyfit(n,freqH2,1);
    FH2=FH2i(1)
    bH2=FH2i(2)

    ni=.1:.1:6;

%    plot(n,freqH1,'r+')
 %   freqiH1=FH1*ni+bH1;
  %  hold on
%    plot(ni,freqiH1,'r-')
 %   hold on

    plot(n,freqH2,'bo')
    freqiH2=FH2*ni+bH2;
    hold on
    plot(ni,freqiH2,'b--')
    hold off

    xlabel('Ordinal number of resonance frequencies')
    ylabel('Frequency (Hz)')
    title('Plot of MATLAB polyfit on 5 Resonance Frequencies on Chan. H1(-
), H2(--)')
    pause
%    mH1=1/2+(2*bH1)/FH1;
    mH2=1/2+(2*bH2)/FH2;

%    nuH1=1/2*(2*mH1+1)/(1+mH1)
    nuH2=1/2*(2*mH2+1)/(1+mH2)

%    ratioH1=(2*FH1)/(1-nuH1)
    ratioH2=(2*FH2)/(1-nuH2)

  end
```

- **r04_polyfit13H2O***.m**

```
clf
clear

filename=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day091_160.frequency'
);
  fid=fopen(filename,'rt');
  f=fscanf(fid,' %f \n');
  fclose(fid);

  for ichunk=1:13
%    ichunk=1
    for ichan=2:3
      if ichan==1

FILE=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day091_160_%2.2d_Z.spectr
a',ichunk)
        fileZ=(FILE);

        fid=fopen(fileZ,'rt');
        Z=fscanf(fid,' %f \n');
        fclose(fid);
        semilogx(f,Z,'r-');
        axis([0.9 70 -165 -75]);
```

```
        TITLE=sprintf('PSD plots Z(-), H1(--), H2(:) of H2O data at FC=160
CHUNK: %d\n',ichunk);
        title(TITLE);
        [freqZ,powerZ]=ginput(13)
%        hold on
      elseif ichan==2

FILE=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day091_160_%2.2d_H1.spect
ra',ichunk)
        fileH1=(FILE);

        fid=fopen(fileH1,'rt');
        H1=fscanf(fid,' %f \n');
        fclose(fid);
        semilogx(f,H1,'r-');
        axis([0.9 70 -185 -95]);
%        [freqH1,powerH1]=ginput(13)
        hold on
      elseif ichan==3

FILE=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day091_160_%2.2d_H2.spect
ra',ichunk)
        fileH2=(FILE);

        fid=fopen(fileH2,'rt');
        H2=fscanf(fid,' %f \n');
        fclose(fid);
        semilogx(f,H2,'b-');
        axis([0.9 70 -185 -95]);
        [freqH2,powerH2]=ginput(13)
        hold off
      end
    end
%    pause


    n=[1 2 3 4 5];

%    freqH1=freqH1';
    freqH2=freqH2';

%    [FH1i,bH1i]=polyfit(n,freqH1,1);
%    FH1=FH1i(1)
%    bH1=FH1i(2)
    [FH2i,bH2i]=polyfit(n,freqH2,1);
    FH2=FH2i(1)
    bH2=FH2i(2)

    ni=.1:.1:13;

%    plot(n,freqH1,'r+')
 %   freqiH1=FH1*ni+bH1;
  %  hold on
%    plot(ni,freqiH1,'r-')
 %   hold on

    plot(n,freqH2,'bo')
    freqiH2=FH2*ni+bH2;
    hold on
    plot(ni,freqiH2,'b--')
    hold off
```

```
    xlabel('Ordinal number of resonance frequencies')
    ylabel('Frequency (Hz)')
    title('Plot of MATLAB polyfit on 13 Resonance Frequencies on Chan. H1(-
), H2(--)')
    pause
%    nuH1=1/2*(2*mH1+1)/(1+mH1)
    nuH2=1/2*(2*mH2+1)/(1+mH2)

%    nuH1=(1+2*mH1)/2*(1+mH1)
    nuH2=(1+2*mH2)/2*(1+mH2)

%    ratioH1=(2*FH1)/(1-nuH1)
    ratioH2=(2*FH2)/(1-nuH2)

  end
```

- **r05_polyfitOSN.m**

```
  n=[1 2 3 4 5];
  freqH=[0.39 0.90 1.21 1.70 2.21];
  [FHi,bHi]=polyfit(n,freqH,1);
  FH=FHi(1);
  bH=FHi(2);

  ni=.1:.1:8;

  plot(n,freqH,'r+')
  freqiH=FH*ni+bH;
  hold on
  plot(ni,freqiH,'r-')
  hold on

  xlabel('Ordinal number of resonance frequencies')
  ylabel('Frequency (Hz)')
  title(' MATLAB polyfit on 5 Resonance Frequencies on Chan. H1(-), H2(--)
of OSN data')
  pause
  mH=1/2+(2*bH)/FH;
  nuH=1/2*(mH+1)/(1+mH)
  ratioH=(2*FH)/(1-nuH)
```

- **r06_pickallrfreqH2O***.m**

```
clf
clear

filename=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day121_160.frquency')
;
  fid=fopen(filename,'rt');
  f=fscanf(fid,' %f \n');
  fclose(fid);

  for ichunk=1:13
%    ichunk=1
    for ichan=2:3
      if ichan==1
```

```
FILE=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day121_%2.2d_C1.spectra',
ichunk)
        fileZ=(FILE);

        fid=fopen(fileZ,'rt');
        Z=fscanf(fid,' %f \n');
        fclose(fid);
        semilogx(f,Z,'r-');
        axis([0.9 70 -165 -75]);
        TITLE=sprintf('PSD plots Z(-), H1(--), H2(:) of H2O data at FC=160
CHUNK: %d\n',ichunk);
        title(TITLE);
        [freqZ(:,ichunk),powerZ(:,ichunk)]=ginput(7);
        %        hold on
      elseif ichan==2

FILE=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day121_%2.2d_C2.spectra',
ichunk)
        fileH1=(FILE);

        fid=fopen(fileH1,'rt');
        H1=fscanf(fid,' %f \n');
        fclose(fid);
        semilogx(f,H1,'r-')
        axis([0.9 70 -165 -75]);
        [freqH1(:,ichunk),powerH1(:,ichunk)]=ginput(30);

      elseif ichan==3

FILE=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day121_%2.2d_C3.spectra',
ichunk)
        fileH2=(FILE);

        fid=fopen(fileH2,'rt');
        H2=fscanf(fid,' %f \n');
        fclose(fid);
        semilogx(f,H2,'b-')
        axis([0.9 70 -165 -75]);
        [freqH2(:,ichunk),powerH2(:,ichunk)]=ginput(30);
        hold off
      end
    end
    save .... f H1 H2 freqH1 freqH2 powerH1 powerH2
  end
```

- **r07_plotallrfreqH2O***.m**

```
clf
clear

    load the30picks ;

%    plot(freqH1,powerH1,'r.');
    semilogx(freqH1,powerH1,'r.');
%    axis([0.9 70 -165 -75]);
    axis([0.9 22 -165 -75]);
    hold on
%    plot(freqH2,powerH2,'b.');
    semilogx(freqH2,powerH2,'b.');
```

```
%    axis([0.9 70 -165 -75]);
    axis([0.9 22 -165 -75]);
    axis([23 44 -165 -75]);
    axis([45 66 -165 -75]);
    A=1:0.2:22
    A=23:0.2:44
    A=45:0.2:66
    axes('XTick',A)
    grid on
    xlabel('Frequency (Hz) ');
    ylabel('psd (dB)');
%    title('Plot of all Frequencie Peaks on Chan. H1(red) of H2O data');
%    title('Plot of all Frequencie Peaks on Chan. H2(bleu) of H2O data');
    title('Plot of all Frequencie Peaks on Chan. H1(red), H2(bleu) of H2O
data');
    hold off
```

- **r08_plotchunkrfreqH2O.m**

```
clf
clear

    load the30picks;

%    plot(freqH1,powerH1,'r.');
    semilogx(freqH1,powerH1,'r.');
%    axis([0.9 70 -165 -75]);
    axis([0.9 22 -165 -75]);
    axis([23 44 -165 -75]);
    axis([45 66 -165 -75]);
    hold on

%    plot(freqH2,powerH2,'b.');
    semilogx(freqH2,powerH2,'b.');
%    axis([0.9 70 -165 -75]);
    axis([0.9 22 -165 -75]);
    axis([23 44 -165 -75]);
    axis([45 66 -165 -75]);
    grid on

    xlabel('Frequency (Hz) ');
    ylabel('psd (dB)');
%    title('Plot of all Frequencie Peaks on Chan. H1(red) of H2O data');
%    title('Plot of all Frequencie Peaks on Chan. H2(bleu) of H2O data');
    title('Plot of all Frequencie Peaks on Chan. H1(red), H2(bleu) of H2O
data');
    hold off

---------------------------------------------------------------------------

A=ones(30,1);
B=A*2;
C=A*3;
D=A*4;
E=A*5;
F=A*6;
G=A*7;
H=A*8;
I=A*9;
J=A*10;
```

```
K=A*11;
L=A*12;
M=A*13;
O=[A B C D E F G H I J K L M];

plot(freqH1,O,'r.');
axis([0.9 22 0 14]);
hold on
plot(freqH2,O,'b.');
axis([45 66 0 14]);
xlabel('Frequency (Hz) ');
ylabel('time(chunks) - 1 chunk = 1.8 hours');
title('All Frequencie Peaks on Chan. H1(red), H2(bleu) of H2O data for
1day');
```

- **r09_plot29rfreqdifH2O***.m**

```
clf
clear

%    load the30picksH2O
    load sortfreq
    N=30;
    for ichunk=1:13
%    ichunk=1
      dfH1(:,ichunk)=sfreqH1(2:N,ichunk)-sfreqH1(1:(N-1),ichunk);
      dfH2(:,ichunk)=sfreqH2(2:N,ichunk)-sfreqH2(1:(N-1),ichunk);
    end
    ni=2:30;
    n=[ni;ni;ni;ni;ni;ni;ni;ni;ni;ni;ni;ni;ni];
    na=n';
    plot(na,dfH1,'r.')
    hold on
    plot(na,dfH2,'b.')
    axis([0 30 0 12.5]);
    ylabel('Frequency Difference (Hz) ');
    xlabel('Ordinal number of freq.peak');
    title('Plot of all Frequencie Difference between Peaks, Chan.
H1(red),H2(bleu)  of H2O data');
    hold off

% I think you need to calculate the differences between all the rpaeks in
% the matrix, and not only between following peaks. How can we do that?
With DIFF ?
```

- **r10_trytick.m**

```
clf
clear

    load [freqH1(:,ichunk),powerH1(:,ichunk)];

    load [freqH2(:,ichunk),powerH2(:,ichunk)];

%    plot(freqH1,powerH1,'r.');
    semilogx(freqH1,powerH1,'r.');
%    axis([0.9 70 -165 -75]);
    axis([0.9 22 -165 -75]);
    hold on
```

```
    plot(freqH2,powerH2,'b.');
%    semilogx(freqH2,powerH2,'b.');
%    axis([0.9 70 -165 -75]);
    axis([0.8 22 -165 -75]);
%    axis([23 44 -165 -75]);
%    axis([45 66 -165 -75]);
    A=0.8:0.2:22;
%    A=23:0.2:44
%    A=45:0.2:66
    axes('XTick',A)
    grid on
    xlabel('Frequency (Hz) ');
    ylabel('psd (dB)');
%    title('Plot of all Frequencie Peaks on Chan. H1(red) of H2O data');
%    title('Plot of all Frequencie Peaks on Chan. H2(bleu) of H2O data');
    title('Plot of all Frequencie Peaks on Chan. H1(red), H2(bleu) of H2O
data');
    hold off
```

- **r11_colorarrayH2O***.m**

```
clf
clear

filename=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day091_160.frequency'
);
  fid=fopen(filename,'rt');
  f=fscanf(fid,' %f \n');
  fclose(fid);

  for ichunk=1:14
%    ichunk=1
    for ichan=1:3
      if ichan==1

FILE=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day091_160_%2.2d_Z.spectr
a',ichunk)
        fileZ=(FILE);

        fid=fopen(fileZ,'rt');
        Z=fscanf(fid,' %f \n');
        fclose(fid);
%         semilogx(f,Z,'r-');
%         axis([0.9 70 -165 -75]);
%         TITLE=sprintf('PSD plots Z(-), H1(--), H2(:) of H2O data at FC=160
CHUNK: %d\n',ichunk);
%         title(TITLE);
        powerallZ(:,ichunk)=Z;
%         hold on

      elseif ichan==2

FILE=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day091_160_%2.2d_H1.spect
ra',ichunk)
        fileH1=(FILE);

        fid=fopen(fileH1,'rt');
        H1=fscanf(fid,' %f \n');
        fclose(fid);
%         semilogx(f,H1,'r-')
```

```
%        axis([0.9 70 -165 -75]);
        powerallH1(:,ichunk)=H1;

      elseif ichan==3

FILE=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day091_160_%2.2d_H2.spect
ra',ichunk)
        fileH2=(FILE);

        fid=fopen(fileH2,'rt');
        H2=fscanf(fid,' %f \n');
        fclose(fid);
%        semilogx(f,H2,'b-')
%        axis([0.9 70 -165 -75]);
        powerallH2(:,ichunk)=H2;
        hold off
      end
    end
  end

  powZ=powerallZ';
  powH1=powerallH1';
  powH2=powerallH2';

  save colorpl091 powZ powH1 powH2;
```

- **r12_colorplotH2O***.m**

```
clf
clear

filename=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day121_160.frequency'
);
  fid=fopen(filename,'rt');
  f=fscanf(fid,' %f \n');
  fclose(fid);
  load colorpl091
  for ichan=2:3
    if ichan==1
      pcolor(powZ(:,643:1576));
      shading('interp');
%     shading('flat');
      title('Variation of amplitude on the high side of the PSD spectrum of
Ch.Z through one day (091) of H2O data');
    elseif ichan==2
      pcolor(powH1(:,643:1576));
      shading('interp');
%     shading('flat');
      title('Variation of amplitude on the high side of the PSD spectrum of
Ch.H1 through one day (091) of H2O data');
    elseif ichan==3
      pcolor(powH2(:,643:1576));
      shading('interp');
%     shading('flat');
      title('Variation of amplitude on the high side of the PSD spectrum of
Ch.H2 through one day (091) of H2O data');
    end
%---------------------------------------------------------------------------
%  Hifreq=find(f>0.9);  %to find the index numbers of the frequencies from
0.9Hz-highest.
```

```
%   Hifreq(1)
% ans = 643
%   Hifreq(end)
% ans = 1576
%------------------------------------------------------------------------

    theLim=[f(643) f(1576)];                   % theLim = 0.9018Hz -
59.8770Hz
    relabel(theLim,'x');
    colorbar;
    xlabel('Frequency (Hz)');
    ylabel('Chunk number, a chunk = 1.8 hours');
    pause
  end

%  would be nice to have tickmarks...
```

- **r13_logcolorplotH2O***.m**

```
clf
clear


filename=sprintf('/osn1/b3s2/OSN/H2O/work/H2O_spectra/day091_160.frequency'
);
  fid=fopen(filename,'rt');
  f=fscanf(fid,' %f \n');
  fclose(fid);

  for ichan=1:3
  ax1=axes('position',[.15 .1 .6 .8]);

  set(ax1,'xlim',[(f(643)) (f(1575))],'xscale','log',...
         'ylim',[1 13],'visible','on','tickdir','out','box','on');

  ax2=axes('position',[.15 .1 .6 .8]);
  load colorpl091
    if ichan==1
      ax2_plot=pcolor(powZ);
      axis tight;

set(ax2,'visible','off','xscale','log','box','on','clim',[min(min(powZ))
max(max(powZ))]);
    elseif ichan==2
      ax2_plot=pcolor(powH1);
      axis tight;

set(ax2,'visible','off','xscale','log','box','on','clim',[min(min(powH1))
max(max(powH1))]);
    elseif ichan==3
      ax2_plot=pcolor(powH2);
      axis tight;

set(ax2,'visible','off','xscale','log','box','on','clim',[min(min(powH2))
max(max(powH2))]);
    end
  shading('interp');
  set(get(ax1,'xlabel'),'string','FREQUENCY log (Hz)');
  set(get(ax1,'ylabel'),'string','CHUNK NUMBER');
  theLim=[log10(f(643)) log10(f(1576))];
```

```
  relabel(theLim,'x');
%  pause
  end
%---------This can be used to make a grid and get a 3D plot-----------------
%   ax3=colorbar;
 % set(ax3,'position',[.85 .1 .05 .8],...
 %     'visible','on','clim',[ min(min(powH1)) max(max(powH1))]);


%  view(45,45)    % This can be used to look at a 3D plot of the data from
different angles)
```

Appendix G
AGU Fall Meeting 2000

December 14 – 19, San Francisco, California, held at the Moscone Convention Centre.
With the help of John Woodside, the VU-fonds and Ralph Stephen I was able to attend this meeting and to present a poster on the subject of "Shear Wave Resonance's in Sediments on the Deep Sea Floor". The presentation was based on the second part of the research concerning the OBS data, which I have done in the summer of 2000 in at WHOI, as described previously in this Report. I authored the poster together with Ralph Stephen and we submitted the following abstract:

Shear Wave Resonance's in Sediments on the Deep Sea Floor

Shear wave resonance's at frequencies between about 0.1 and 10 Hz are a ubiquitous feature of ambient noise and controlled source seismic data acquired on sedimented sea floors. They are a major factor in the ambient noise field and mask many useful seismic arrivals. For controlled source experiments shear wave resonance's are a major source of incoherent, signal generated noise and coda. The peaks of the ambient noise spectra associated with the resonance's, however, can be used to infer the sediment rigidity and thickness. The theory of Godin and Chapman (1999) has been used to infer shear velocity and sediment thickness from the resonance peaks in horizontal component power spectra for two sites in the Pacific. At ODP Site 843B (OSN-1), about 225 km southwest of Oahu, the sediment thickness is known from drilling and we can infer from the resonance's that the uppermost shear velocity is 76m/s. At the Hawaii-2 Observatory (H2O) Site, in 5000m water depth half-way between Hawaii and California we predict a sedimentary thickness of about 50m by assuming the same uppermost shear velocity as at OSN-1.

The poster presentation can be viewed at http://msg.whoi.edu/msg.html

I presented the poster on Friday the 15th of december in the big D-hall. There where several hundreds of poster- and oral presentations every day on various subjects like Education and Human Resources, Atmospheric Sciences, Biogeosciences, Geodesy, Geomagnetism and Paleomagnetism, Hydrology, Ocean Sciences, Planetary Sciences, Seismology Tectonophysics, Volcanology, Geochemistry and Petrology.
There was a lot of interest in my poster and I discussed various features of the model with various people. Some questions that where: *What does the total velocity curve look like? What is the shear wave velocity at the bottom of the sediment layer?* (G.H.Sutton) *Are elastic properties used in this model?* (P.Spudich) Most people where very surprised (either positively or negatively) about the small sedimentthickness we concluded for the H2O Site and also about the low shear velocities we constrained for the OSN Site. I told them that this was still work in progress and explained some of the assumptions we have made and how big their influence was in the final picture. I spoke to many Japanese scientists who where very interested and were doing similar research. Sometimes I needed Ralph Stephen's help to explain some of the more complicated geophysical techniques and formula's used. The following people were all very interested and would like to have a copy of my final report sent to them:

S. Sutton	ghsutton@aol.com	
John Osler		Defense Res. Establ. Atlantic
Dave Chapman		
Kimihiro Mochizuki	kimi@ori.u-tokyo.ac.jp	Ocean Research Inst. University of Tokyo
Paul Spudich		USGS MS?22 345 Middlefield Rd Menlo Park CA 94025
Brian Bonner	bonner1@llnl.gow	L-201 LLNL Livermore CA 94550
Eiichiro Araki	araki@jamstec.go.jp	Japan Marine Science and Technology Center (JAMSTEC) Dep. Deep Sea Research
P.Bromirski	peter@coast.ucsd.edu	Sripps Institution of Oceanography (Geophysics) San Diego
John N.Louie	louie@seismo.unr.edu	Associate Prof. Of Seismology Seismological Laboratory Dep. Of Geological Sciences Mackay School of Mines Nevada
Frank Scherbaum	fs@geo.uni-potsdam.de	Universitat Potsdam Mathematisch-Naturwissenschfliche Fakultat Institut fur Geowissensch.

During the rest of the Meeting I attended lectures on several topics, most of them about either Tectonics or Geophysics, but also some about Planetary Sciences and Climatology which I think are subjects of general interest. I also looked at posters and met some people from Amsterdam, one student who was in my year at university is now doing a PhD at Scripps and he introduced me to other people working there. Another former VU student who is now doing a post-doc at the Berkeley Laboratory in CA. I talked a lot to PhD students that I already knew or met at the meeting and learned more about their research and how they liked being a PhD student. Personally I am still in doubt about what I want to do after I graduate (stay in science or work for a company) and talking to these people and being at this congress gave me a chance to learn more about doing research. On the other hand, there was also a big hall with representatives of many companies and research instituations from the USA and some other countries, wich was very interesting. Some Institutions and companies represented here were ODP, USGS, Atmospheric Sciences Data Center, NASA, Digital Technology Associates (who delivered and constructed the Guralp seismometers used at the H2O and OSN sites), the Space Studies Board, Geometrics (maker of the on-board acquisition and processing equipment) and many others. I thought it was a great experience!!!

Acknowledgements

I would like to thank in the first place John Woodside (does great research at the Vrije Universiteit in the field of Marine Geology). He has motivated me to put my best into this research. He has put me in touch with Ralph Stephen at WHOI and made it possible through personal and financial assistance that I was able to write this report and attend the AGU Fall Meeting in San Francisco in December 2000.
Two other people who I am very grateful of having had the opportunity to work with are Ralph Stephen and Tom Bolmer of Woods Hole Oceanographic Institution. Ralph invited me to come over and work on Broadband Seismometer Data. He gave me a good solid guidance when I sometimes tended to drift of the straight course to the details of the research. He tried to explain some geophysics to a structural geologist and came a long way. Tom assisted me in all my computer trials (and many errors) both at WHOI and back at the VU and was a friendly colleague for some months. Without these people there would be no Research Report.

I have to thank the VU-fonds and WHOI for giving me financial support during my stay at WHOI and during the AGU Meeting.

Finally, I would like to thank my family for supporting me spiritually, myself for working hard and persevering with this research in times when it was difficult, and the Earth for being such a fascinating place to live on and study.

References

Abramowitz, M., and I.A.Stegun, Eds.
Handbook of mathematical functions with formulas, graphs, and tables
Appl. Math. Ser., Vol.55 (national Bureau of Standards), Washington, 1964

Adair, R. G., J. A. Orcutt, and T.H. Jordan
Analysis of ambient seismic noise recorded by downhole and oceanbottom seismometers on deep sea drilling project leg 78B
Initial Rep. Deep Sea Drill. Proj., 78, 767-780, 1984

Agnew, D., J. Berger, R. Buland, W. Farrel, and F. Gilbert
International deployment of accelerometers: A network for very long period seismology
Eos, Transactions, Am. Geophys. Union, 57 (4), 180-187, 1976

Babcock, J. M., B. A. Kirkendall, and J. A. Orcutt
Relationships between ocean bottom noise and the environment
Bull. Seismol. Soc. Am., 84(6), 1991-2007, 1994

Bendat, J.S., and A.G. Piersol
Random Data: Analysis and measurement procedures
566pp., John Wiley and Sons, New York, 1986

Blackman, D.K., J. A. Orcutt, and D.W. Forsyth
Teleseismic detection using ocean bottom seismometers at mid-ocean ridges
Bull. Seismol. Soc. Am., 85(6), 16481664, 1995

Bolt, B. A.
Nuclear explosions and Earthquakes: The Parted Veil
W. H. Freeman, New York, 1976

Bradley, C.R., R. A. Stephen, L. M. Dorman, and J. A. Orcutt
Very low frequency (0.2 – 10 Hz) seismoacoustic noise below the seafloor
J. Geophys. Res., 102 (B6), 11,703 – 11,718, 1997

Brekhovskikh, LM., and O.A. Godin
Acoustics of layered medium I: plane and quasi-plane waves
Springer-Verlag, Berlin, 1990 and 1998

Brink, K. H.
Tidal and lower frequency currents above Fieberling Guyot
J. Geophys. Res., 100 (C6), 10817-10832, 1995

Butler, R., C. S. McCreery, L. N. Frazer, and D. A. Walker
High frequency seismic attenuation of oceanic P and S waves in the western Pacific
J. Geophys. Res., 92(B2), 1383 – 1396, 1987

Chapman, N.R., and D.M.F. Chapman
A coherent ray model of plane-wave reflection from a thin sediment layer
J. Acoust. Soc. Am. 94, 2731-2738, 1993

Chave, A. D.., D. S. Luther, and J. H. Filloux
The barotropic electromagnetic and pressure experiment, 1, Barotropic response to atmospheric forcing
J. Geophys. Res., 97(C6), 9565 – 9593, 1992

Collins, J.A., F.L. Vernon, J.A. Orcutt, R.A. Stephen, J.R. Peal, J.A. Hildebrand, and P.N. Spiess

Relative performance of the borehole, surficially buried, and seafloor broadband seismographs on the Ocean Seismic Network pilot experiment: frequency-domain results
EOS, Transactions AGU, 79 (45), 661, 1998

Cox, C.S., T.Deaton, and S.C. Webb
A deep sea differntial pressure gauge
J. Atmos. Oceanic Technol., 1, 237-246, 1984.

Crawford, W. C., S. C. Webb, and J. A. Hildebrand
Seafloor complience observed by long period pressure and displacement measurements
J. Geophys. Res., 96, 16,151 – 16,160, 1991

Crawford, W.C., S.C. Webb, and Hildebrand
Constraints on melt in the lower crust and MOHO at the East Pacific Rise, 9°48'N, using seafloor compliance measurements
J. Geophys. Res., 104 (2), 2923-2939, 1999

Duennebier, F. K., and G. H. Sutton
Fidelity of ocean bottom seismometers
Mar. Geophysics Res. , 17(6), 535-555, 1995

Duennebier, F. K., G. Blankington, and G. H. Sutton
Current generated noise recorded on ocean bottom seismometers
Mar. Geophysics Res., 5, 109-115, 1981

Duennebier, F. K., D. Harris, S. Poulos, and B.
A buried ocean bottom seismometer: Results from the ULF/VLF experiment (abstract)
Eos Trans. AGU, 72(44), Fall Meeting Suppl., 303, 1991

Duennebier, F.K., R. Butler, A. Chave, D. Harris, J. Jolly, J. Babinec, C. Wolfe
Broad-band seismograms from the Hawaii-2 Observatory
A AGU poster presentation, S.O.E.S.T., University of Hawaii, I.R.I.S., W.H.O.I.

Filloux, J. H.
Pressure fluctuations on the open ocean floor over a broad frequency range: New program and early results
J. Phys. Oceanogr., 10 (10), 1959-1971, 1980

Forsyth, D. W.
Partial melting beneath a Mid-Atlantic ridge segment detected by teleseismic PKP arrivals
Geophys. Res. Lett., 23(5), 463-466, 1996

Godin, O.A., D. M. F. Chapman
Shear-speed gradients and ocean seismo-acoustic noise resonance's

Gross, T. F., A. J. Williams III, and W. D. Grant
Long-term in-situ calculations of kinetic energy and Reynolds stress in a deep sea boundary layer
J. Geophys. Res., 91(C7), 8461-8469, 1986

Hall, M.V.,
Acoustic reflectivity of a sandy seabed: A semianalytic model of the effect of coupling due to shear modulus profile
J. Acoustic Soc. Am. 98, 1075-1079, 1995

Hamilton, E.I.,
Attenuation of shear waves in marine sediments
J. Acoust. Soc. Am. 60, 334-338, 1976

Hasselman, K.

A statistical analysis of the generation of microseisms
Rev. Geophys., 1, 177-209, 1963

Hedlin, M. A., and J. A. Orcutt
A comparative study of island, seafloor and subseafloor ambient noise levels
Bull. Seismol. Soc. Am., 79(1), 172-179, 1989

Hovem, J.M., M.D. Richardson, and R.D. Stoll, Eds.
Shear waves in marine sediments
Kluwer, Dordrecht, 1991

Hughes, S.J., D.D. Ellis, D.M.F. Chapman, and P.R. Staal
Low-frequency acoustic propagation loss in shallow water over hard-rock seabeds covered by a thin layer of elastic-solid sediment
J. Acoustic Soc. Am. 88, 283-297, 1990

Iwasaki, T., and F. Tatsuoka
Effects of grain size and grading on dynamic shear moduli of sands
Soils Found. 17, 19-35, 1997

Kibblewhite, A.C., and C.Y. Wu
Wave Interactions as a seismo-acoustic source
Springer-Verlag, Berlin, 1996

Lacoss, R. T., E. J. Kelly, and M. N. Toksoz
Estimation of seismic noise structure using arrays
Geophysics, 34(1), 21-38, 1969

Lacoste, L.J.B.
Measurement of gravity at sea and in the air
Rev. Geophys., 5, 477-526, 1967

Li, Y., W. Prothero, C. Thurber, and R. Butler
Observations of ambient noise and coherency on the island of Hawaii for teleseismic studies
Bull. Seismol. Soc. Am., 84(4), 1229-1242, 1994

Lighthill, J.
Waves in Fluids
p.504, Cambridge Univ. Press, New York, 1979

McCreery, C. S., F. K. Duennebier, and G. H. Sutton
Correlation of deep ocean noise (0.4-20Hz) with wind, and the Holu spectrum – a worldwide constant
J. Acoust. Soc. Am., 93(5), 2639-2648, 1993

Montagner, J.-P.,J.-F. Karczewski, B. Romanowitz, S. Bouaricha et al
The French pilot experiment OFM-SISMOBS; first scientific results on noise level and event detection
Phys. Earth Plan. Int., 84 (1-4), 321-336, 1994

Osler, J.C., and D.M.F. Chapman
Seimo-acoustic determination of the shear-wave speed of surficial clay and silt sediments on the Scotian Shelf
Can. Acoust. 24, 11-22, 1996

Peterson, J.,
Observations and modelling of seismic background noise
U. S. Geol. Surv. Open File Rep., 93-322, 1993.

Purdy, G. M., and A. D. Dziewonski
Proceedings of a workshop on Broad Band Seismometers in the Deep Ocean
331 pp., WHOI, Woods Hole, Mass., 1988

Rhines, P. B.
The dynamics of unsteady currents
The Sea, vol. 6, Marine Modelling, pp. 189-318, John Wiley, New York, 1977

Romanowicz, B.
Seismic tomography of the Earth's mantle
Annu. Rev. Earth Planet. Sci., 19, 77-99, 1991

Sato, T., K. Katsumata, J. Kasahara, N. Hirata, R. Hino, N. Takahashi, M. Sekine, S. Miura, and S. Koresawa
Travel time residuals of teleseismic P-waves at the Rodriguez triple junction in the Indian Ocean using ocean bottom seismometers
Geophys. Res. Lett., 23(7), 713-716, 1996

Scholz, C. H.
The mechanics of earthquakes and faulting
439 pp., Cambridge University Press, New York, 1990

Schreiner A. E., and L. M. Dorman
Coherence lengths of seafloor noise: Effect of ocean bottom structure
J. Acoust. Soc. Am., 88(3), 1503-1514, 1990

Snodgrass, F. E., G. W. Groves, K. F. Hasselman, G. R. Miller, W. H. Munk, and W. H. Powers
Propagation of ocean swell across the Pacific
Philos. Trans. R. Soc. London, Ser. A, 259(1103), 431-479, 1966

Stephen, R. A., et al.
The seafloor borehole seismic system (SEABASS) and ULF noise
Mar. Geophys. Res., 16, 243-269, 1994

Stephen, R. A., J. H. Natland, R. Butler, K. Becker, A. D. Chave and F. K. Duennebier
Drilling fast spread Pacific Crust at the H2O long term seafloor observatory
JOIDES/Ocean Program- Drilling Proposal, 1997

Stephen, R.A., J.A. Collins, J.A. Hildebrand, J.A. Orcutt, K.R. Peal, F.N. Spiess, and F.L. Vernon
Broadband borehole seismology and real-time submarine observatories
IRIS, 1998

Stephen, R.A., J.A. Collins, J.A. Hildebrand, J.A. Orcutt, K.R. Peal, F.N. Spiess, F.L. Vernon
A pilot experiment for broadband seismic measurements on the deep ocean floor
Report for Cruise TN074 on the R/V Thomas G. Thompson

Sutton, G. H., W. G. McDonnel, D. D. Prentiss, and S. N. Thanos
Ocean bottom seismic observatories
Proc. IEEE, 53, 1909-1921, 1965

Sutton, G. H., and N. Barstow
Ocean bottom ultra low frequency (ULF) seismo-acoutic ambient noise: 0.002-0.04 Hz
J. Acoust. Soc. Am., 87, 2005-2012, 1990

Thomson, R.E., S.E. Roth, and J. Dymond
Near inertial motions over a mid-ocean ridge:effects of topography and hydrothermal plumes
J. Geophys. Res., 95 (C5), 7261-7278, 1990

Trehu, A. M.
A note on the effect of bottom currents on a ocean bottom seismometer
Bull. Seismol. Soc. Am., 75(4), 1195-1204, 1985b

Trehu, A. M., and S. C. Solomon
Coupling parameters of the MIT OBS at two nearshore sites
Mar. Geophys. Res., 5, 68-78, 1981

Walker, D. A., C. S. McCreery, and G. H. Sutton
Spectral characteristics of high-frequency Pn, Sn phases in the western Pacific
J. Geophys.Res., 88(B5), 4289-4298, 1983

Webb, S. C.
Long period acoustic and seismic measurements and ocean floor currents
IEEE J. Oceanic Eng., 13, 263-270, 1988

Webb, S. C., X. Zhang, and W. C. Crawford
Infragravity waves in the deep ocean
J. Geophys. Res., 96(C2), 2723-2736, 1991

Webb, S. C., and A. D. Schultz
Very low frequency ambient noise at the seafloor under the Beafort Sea icecap
J. Acoust. Soc. Am., 91(3), 1429-1439, 1992

Webb, S.C. – Scripps Institution of Oceanography, University of California, San Diego, La Jolla
Broadband seismology and noise under the ocean
Reviews of Geophysics, 36, 1 / February 1998, pages 105-142, paper nr.97RG02287

Webb, S.C., and W.C. Crawford
Long period seafloor seismology and deformation under ocean waves
Bull. Seis. Soc. Am., 89 (6), 1535-1542, 1999

Wenz, G. M.
Acoustic ambient noise in the ocean: Spectra and sources
J. Acoust. Soc. Am., 34(12), 1936-1956, 1962

White, J.E.
Underground sound application of seismic waves
Elsevier, Amsterdam, 1983, pp. 67-68

Wysession, M. E.
How well do we utilize global seismicity?
Bull. Seismol. Soc. Am., 86(5), 1207-1219, 1996

Yamamoto, T., M. V. Treverrow, M. Badley, and A. Turgut
Determination of seabed porosity and shear modulus profiles using a gravity wave inversion
Geophys. J. Int., 98, 173-182, 1989

Zhang, J., and C. A. Langston
Constraints on oceanic lithosphere from deep focus regional earthquake receiver function inversions
J. Geophys. Res., 100(B11), 22,178-22,196, 1995